Package ‘allcontributors’

December 2, 2020

Title Acknowledge all Contributors to a Project
Version 0.0.2
Description Acknowledge all contributors to a project via a single function call. The function appends to a ‘README’ or other specified file(s) a table with names of all individuals who contributed via code or repository issues. The package also includes several additional functions to extract and quantify contributions to any repository.
License GPL-3
URL https://github.com/ropenscilabs/allcontributors
BugReports https://github.com/ropenscilabs/allcontributors/issues
Imports cli, clipr, ghql, git2r, htr, jsonlite, magrittr
Suggests knitr, rmarkdown, testthat, webmockr
VignetteBuilder knitr
Encoding UTF-8
Language en-GB
LazyData true
RoxygenNote 7.1.1
NeedsCompilation no
Author Mark Padgham [aut, cre]
Maintainer Mark Padgham <mark.padgham@email.com>
Depends R (>= 3.5.0)
Repository CRAN
Date/Publication 2020-12-02 10:50:06 UTC

R topics documented:

add_contributors ....................................................... 2
get_contributors ....................................................... 3
add_contributors

Description
Add contributors to README.Rmd

Usage
add_contributors(
  repo = ".",
  ncols = 7,
  files = c("README.Rmd", "README.md"),
  type = c("code", "issues", "discussion"),
  num_sections = 3,
  section_names = c("Code", "Issue Authors", "Issue Contributors"),
  format = "grid",
  alphabetical = FALSE,
  open_issue = FALSE,
  force_update = FALSE
)

Arguments
repo Location of repository for which contributions are to be extracted. This must be
git project with a github remote.
ncols Number of columns for contributors in 'README'
files Names of files in which to add contributors
type Type of contributions to include: 'code' for direct code contributions (including
documentation), 'issues' to recognise contributors who open issues, and 'discussion' for contributing to discussions within issues. Discussion contributions are only from individuals not present in either 'issues' or 'code'; and 'issues' contributions are only from individuals not present in 'code'.
num_sections Number of sections in which to divide contributors:
  • 1 All contributions within single section regardless of type
  • 2 Contributions divided between a single section for code and a second
    section for all other issue-related contributions.
  • 3 Contributions divided into single sections for each of the three type ar-
    guments.
get_contributors

section_names Names of the sections to appear on the nominated files.

format One of ("grid", "list", "text") to control display of contributors as

• 1 "grid" for a rectangular grid, with each contributor represented by their
github avatar, with avatar linked to contributor profile and name linked to
repository contributions.
• 2 "list" for a more condensed list with github user names only and no
avatars, one contributor per line linked to issue contributions.
• 3 "text" for a single line of text containing comma-separated github user
names linked to issue contributions.

alphabetical If TRUE, order contributors alphabetically, otherwise order by decreasing num-
bers of contributions.

open_issue If TRUE, open or edit an issue on github in order to notify all contributors that
they’ve been added to your README (see Note).

force_update If TRUE, update the specified files even if contributions have not changed.

Value

Named list of logical values indicating whether files of given names were updated or not is returned
invisibly (that is, only if explicitly assigned to a return value).

Note

Opening an issue on github requires the github command-line interface to be locally installed. See
https://cli.github.com/.

Examples

# The following code extracts the contributors from the git repository
# associated with current working directory and writes them to a file.
## Not run:
f <- tempfile (fileext = ".Rmd")
writeLines ("", f) # blank file in tempdir()
add_contributors (repo = ".", files = f)
## End(Not run)

Description

Get all contributors to a repository, including those who contribute to code, open issues, and con-
tribute to discussions in issues.
get_gh_code_contributors

Usage

get_contributors(
  org,
  repo,
  type = c("code", "issues", "discussion"),
  alphabetical = FALSE,
  quiet = FALSE
)

Arguments

org Github organisation name for repository
repo Repository within org for which contributors are to be extracted
type Type of contributions to include: 'code' for direct code contributions (including
documentation), 'issues' to recognise contributors who open issues, and 'discussion' for contributing to discussions within issues. Discussion contributions are only from individuals not present in either 'issues' or 'code'; and 'issues' contributions are only from individuals not present in 'code'.
alphabetical If TRUE, order contributors alphabetically, otherwise order by decreasing numbers of contributions.
quiet If FALSE, display progress information on screen.

Examples

## Not run:
get_contributors (org = "ropenscilabs", repo = "allcontributors")

## End(Not run)

get_gh_code_contributors

Description

Get list of all code contributors to the code of a repository

Usage

get_gh_code_contributors(org, repo, alphabetical = FALSE)

Arguments

org Github organisation name for repository
repo Repository within org for which contributors are to be extracted
alphabetical If TRUE, order contributors alphabetically, otherwise order by decreasing numbers of contributions.
**get_gh_contrib_issue**

**Value**

A *data.frame* of two columns of contributor (name, login)

**Examples**

```r
## Not run:
get_gh_code_contributors (org = "ropenscilabs", repo = "allcontributors")
## End(Not run)
```

**Description**

Extract contributors currently listed on an "All Contributions" issue in a github repository.

**Usage**

```r
get_gh_contrib_issue(org, repo)
```

**Arguments**

- `org`  
  Github organisation name for repository
- `repo`  
  Repository within `org` for which contributors are to be extracted

**Value**

Character vector of github logins for all contributors listed in current issue, or empty character string if there no issue named "All Contributors".

**Examples**

```r
## Not run:
get_gh_contrib_issue (org = "ropenscilabs", repo = "allcontributors")
## End(Not run)
```
**get_gh_issue_people**

*Description*

Extract lists of (1) all authors of, and (2) all contributors to, all github issues for nominated repository.

*Usage*

`get_gh_issue_people(org, repo)`

*Arguments*

- `org` Github organisation name for repository
- `repo` Repository within `org` for which contributors are to be extracted

*Value*

List of (authors, contributors), each as character vector of github login names.

*Examples*

```r
## Not run:
get_gh_issue_people (org = "ropenscilabs", repo = "allcontributors")
## End(Not run)
```

**get_gh_issue_titles**

*Description*

Extract titles and numbers of all issues associated with a nominated repository.

*Usage*

`get_gh_issue_titles(org, repo)`

*Arguments*

- `org` Github organisation name for repository
- `repo` Repository within `org` for which contributors are to be extracted

*Value*

`data.frame` with one column of issue numbers, and one column of issue titles.
Examples

```r
## Not run:
get_gh_issue_titles (org = "ropenscilabs", repo = "allcontributors")

## End(Not run)
```
Index

add_contributors, 2
get_contributors, 3
get_gh_code_contributors, 4
get_gh_contrib_issue, 5
get_gh_issue_people, 6
get_gh_issue_titles, 6