## Package ‘rworkflows’

September 14, 2023

**Type** Package

**Title** Test, Document, Containerise, and Deploy R Packages

**Version** 0.99.13

**Description** Reproducibility is essential to the progress of research, yet achieving it remains elusive even in computational fields. Continuous Integration (CI) platforms offer a powerful way to launch automated workflows to check and document code, but often require considerable time, effort, and technical expertise to setup. We therefore developed the rworkflows suite to make robust CI workflows easy and freely accessible to all R package developers. rworkflows consists of 1) a CRAN/Bioconductor-compatible R package template, 2) an R package to quickly implement a standardised workflow, and 3) a centrally maintained GitHub Action.

**URL** [https://github.com/neurogenomics/rworkflows](https://github.com/neurogenomics/rworkflows), [https://CRAN.R-project.org/package=rworkflows](https://CRAN.R-project.org/package=rworkflows)

**BugReports** [https://github.com/neurogenomics/rworkflows/issues](https://github.com/neurogenomics/rworkflows/issues)

**Encoding** UTF-8

**biocViews** WorkflowManagement

**Depends** R (= 4.1)

**Imports** stats, here, yaml, utils, desc, badger, renv, tools, methods, BiocManager, data.table

**Suggests** markdown, rmarkdown, remotes, knitr, testthat (= 3.0.0), htmltools, jsonlite, BiocStyle, BiocPkgTools, biocViews

**VignetteBuilder** knitr

**License** GPL-3

**Config/testthat/edition** 3

**LazyData** true

**RoxygenNote** 7.2.3.9000

**NeedsCompilation** no

**Author** Brian Schilder [aut, cre] (<https://orcid.org/0000-0001-5949-2191>), Alan Murphy [aut, ctb] (<https://orcid.org/0000-0002-2487-8753>), Nathan Skene [aut] (<https://orcid.org/0000-0002-6807-3180>)


Maintainer  Brian Schilder <brian_schilder@alumni.brown.edu>
Repository  CRAN
Date/Publication  2023-09-14 17:30:02 UTC

R topics documented:

biocpkgtools_db .................................................. 2
bioc_r_versions .................................................. 3
construct_authors ............................................... 4
construct_cont .................................................. 4
construct_runners ............................................... 5
dt_to_desc ....................................................... 7
fill_description .................................................. 7
get_description .................................................. 10
get_hex ............................................................ 11
infer_biocviews .................................................. 12
infer_deps ......................................................... 13
is_gha .............................................................. 14
use_badges ......................................................... 15
use_dockerfile .................................................... 17
use_issue_template ............................................... 18
use_readme ........................................................ 19
use_vignette_docker ............................................ 19
use_vignette_getstarted ....................................... 21
use_workflow ...................................................... 22

Index 25

biocpkgtools_db  Static Bioconductor packages list

Description

A static snapshot of all Bioconductor packages from biocPkgList. Last updated: Sept. 06 2023

Usage

data("biocpkgtools_db")

Format

An object of class data.frame (inherits from data.frame) with 100 rows and 53 columns.
bioc_r_versions

Source

as_ascii <- function(dt, cols=names(dt)){
  cols <- cols[cols func <- function(v){
    Encoding(v) <- "latin1"
    iconv(v, "latin1", "UTF-8")
  } for(col in cols){
    if(is.character(dt[[col]])){
      dt[[col]] <- func(dt[[col]])
    }
  }
  return(dt)
}

biocpkgtools_db <- get_description_repo_biocpkgtools(repo="BioCsoft")
biocpkgtools_db <- as_ascii(biocpkgtools_db[seq(100)])
usethis::use_data(biocpkgtools_db, overwrite = TRUE)

bioc_r_versions  Bioconductor / R versions

Description

Get the respective version of R for a given version of Bioconductor.

Usage

bioc_r_versions(bioc_version = NULL, depth = NULL, return_opts = FALSE)

Arguments

bioc_version  Version of Bioc to return info for. Can be:
  • "devel"Get the current development version of Bioc.
  • "release"Get the current release version of Bioc.
  • <numeric>A specific Bioc version number (e.g. 3.16).
  • NULLReturn info for all Bioc versions.

depth  How many levels deep into the R version to include. For example, is the R version number is "4.2.0", the following depths would return:
  • depth=NULL: "4.2.0"
  • depth=1: "4"
  • depth=2: "4.2"
  • depth=3: "4.2.0"

return_opts  Return a character vector of all valid Bioc version names.

Value

Named list of Bioc/R versions

Examples

ver <- bioc_r_versions(bioc_version="devel")
construct_authors  Construct authors

Description

Helper function to construct an author list for a DESCRIPTION file. Returns a template when authors is not provided (default).

Usage

construct_authors(
  authors = NULL,
  template = c(person(given = "yourGivenName", family = "yourFamilyName", role = "cre", email = "yourEmail@email.com", comment = c(ORCID = "yourOrcidId"))
)
)

Arguments

- authors  A list of authors who contributed to your R package, each provided as objects of class person. By default, if an Authors field already exists in the DESCRIPTION file, the original values are kept. Otherwise, a template person list is created using the construct_authors.
- template  Default value to return when authors=NULL.

Value

Named list in person format.

Examples

authors <- construct_authors()

construct_cont  Construct containers list

Description

Construct containers list
construct_runners

Usage

construct_cont(
  default_cont = "bioconductor/bioconductor_docker",
  default_tag = "devel",
  cont = list(paste(default_cont, default_tag, sep = ":"), NULL, NULL),
  versions_explicit = FALSE,
  run_check_cont = FALSE,
  verbose = TRUE
)

Arguments

default_cont The DockerHub container to default to. Used when it’s detected that only the tag has been given in one or more cont entry.
default_tag The DockerHub container tag to default to.
cont Which Docker container to use on each OS (NULL means no container will be used for that OS). See here for a list of all official Bioconductor Docker container versions.
versions_explicit Specify R/Bioc versions explicitly (e.g. r: 4.2.0, bioc: 3.16) as opposed to flexibly (e.g. r: "latest", bioc: "release").
run_check_cont Check whether the requested container repo (and the tag, if specified) exist using check_cont.
verbose Print messages.

Value

Named list of containers

Examples

cont <- construct_cont()

Description

Construct runner configurations across multiple Operating Systems (OS) for GitHub Actions workflow.
Usage

```r
construct_runners(
  os = c("ubuntu-latest", "macOS-latest", "windows-latest"),
  bioc = list("devel", "release", "release"),
  r = list("auto", "auto", "auto"),
  versions_explicit = FALSE,
  run_check_cont = FALSE,
  cont = construct_cont(default_tag = bioc[[1]], run_check_cont = run_check_cont),
  rspm = list(paste0("https://packagemanager.rstudio.com/",
                  "cran/\_\_linux\_/latest/release"), NULL, NULL),
  verbose = TRUE
)
```

Arguments

- `os` Which OS to launch GitHub Actions on.
- `bioc` Which Bioconductor version to use on each OS. See `bioc_r_versions` documentation for all options.
- `r` Which R version to use on each OS.
- `versions_explicit` Specify R/Bioc versions explicitly (e.g. r: 4.2.0, bioc: 3.16) as opposed to flexibly (e.g. r: "latest", bioc: "release").
- `run_check_cont` Check whether the requested container repo (and the tag, if specified) exist using `check_cont`.
- `cont` Which Docker container to use on each OS (NULL means no container will be used for that OS). See here for a list of all official Bioconductor Docker container versions.
- `rspm` Which R repository manager to use on each OS (NULL means the default will be used for that OS).
- `verbose` Print messages.

Value

Named list of configurations for each runner OS.

Examples

```
runners <- construct_runners()
```
**dt_to_desc**  

**data.table to desc**

---

**Description**

Convert data.table containing the parsed DESCRIPTION file data and convert each of them to desc format.

**Usage**

```r
dt_to_desc(db, refs = NULL, verbose = TRUE)
```

**Arguments**

- `db`  
  A data.table where each row is a different R package and each column is a field from the DESCRIPTION file.

- `refs`  
  Reference for one or more GitHub repository in owner/repo format (e.g. "neurogenomics/rworkflows"), or an R package name (e.g. "rworkflows").

- `verbose`  
  Print messages.

**Value**

A named list of desc objects.

**Examples**

```r
#### Updated data ####
# db <- BiocPkgTools::biocPkgList()
#### Static data ####
db <- rworkflows::biocpkgtools_db
dl <- dt_to_desc(db=db, refs="ABSSeq")
```

---

**fill_description**  

**Fill DESCRIPTION**

---

**Description**

Fill out a DESCRIPTION file, such as (but not limited to) the one provided by the templateR R package template. For any given field, set its corresponding argument as follows to get certain behaviour:

- **NULL**: Keeps the current value.
- **NA**: Removes the field from the DESCRIPTION file entirely.
Usage

fill_description(
    path = here::here("DESCRIPTION"),
    package,
    title,
    description,
    github_owner = NULL,
    github_repo = package,
    authors = construct_authors(authors = NULL),
    depends = paste0("R ", ">= ", bioc_r_versions(bioc_version = "devel", depth = 2)$r,
                     "")",
    imports = infer_deps(which = "Imports", add_newlines = TRUE),
    suggests = infer_deps(which = "Suggests", add_newlines = TRUE),
    remotes = NULL,
    version = NULL,
    license = NULL,
    encoding = NULL,
    vignettebuilder = NULL,
    biocviews = infer_biocviews(pkgdir = dirname(path), add_newlines = TRUE),
    url = paste0("https://github.com/", github_owner, "/", github_repo),
    bugreports = paste0(url, "/issues"),
    save_path = path,
    verbose = TRUE,
    fields = list()
)

Arguments

path Path to the DESCRIPTION file.
package The name of your R package.
title The title of your R package.
description The description of your R package.
github_owner The owner of your R package’s GitHub repository. Can be inferred from the URL field in the DESCRIPTION file if this has already been filled out.
github_repo The name of your R package’s GitHub repository.
authors A list of authors who contributed to your R package, each provided as objects of class person. By default, if an Authors field already exists in the DESCRIPTION file, the original values are kept. Otherwise, a template person list is created using the construct_authors.
depends R package Depends. Defaults to the version of R that the current development version of Bioconductor depends on.
imports R package Imports. These dependencies will be automatically installed with your R package.
suggests R package Suggests. These dependencies will NOT be automatically installed with your R package, unless otherwise specified by users during installation
fill_description

remotes R package Remotes
version The current version of your R package (e.g 0.99.0).
license R package license. See here for guidance.
encoding R package Encoding.
vignettebuilder R package VignetteBuilder.
biocviews Standardised biocViews terms used to describe your package. Defaults to automatically recommending terms using the infer_biocviews function. Note that non-Bioconductor packages (e.g. CRAN) can also use this field.
url URL where your R package is distributed from (e.g. GitHub repository, Bioconductor page, and/or CRAN page). Can be a single character string or a character vector.
bugreports A URL where users of your package should go if they encounter bugs or have feature requests.
save_path Path to save the updated DESCRIPTION file to. Defaults to overwriting the input file (path). Set to NULL if you wish to only return the description object without writing to any file.
verbose Print messages.
fields A named list of additional fields to fill the DESCRIPTION file with: e.g. list(RoxygenNote=7.2.3)

Value
An object of class description.

Examples

#### Get example DESCRIPTION file ####
url <- "https://github.com/neurogenomics/templateR/raw/master/DESCRIPTION"
path <- tempfile(fileext = "DESCRIPTION")
utils::download.file(url,path)

#### Fill out DESCRIPTION file ####
d <- fill_description(
  path = path,
  package = "MyPackageName",
  title = "This Package Does Awesome Stuff",
  description = paste(
    "MyPackageName does several awesome things."
  ),
  github_owner = "OwnerName",
  biocviews = c("Genetics", "SystemsBiology"))
get_description  

Get DESCRIPTION

Description

The **Liam Neeson** of `DESCRIPTION` file functions.

1. I will look for you,
2. I will find you,
3. —and I will import you into a neatly parsed R object.

Uses a variety of alternative methods, including searching locally and on GitHub (whenever possible). Prioritises the fastest methods that do not involve downloading files first.

Usage

```r
get_description(
  refs = NULL,
  paths = here::here("DESCRIPTION"),
  db = NULL,
  cache_dir = tools::R_user_dir(package = "rworkflows", which = "cache"),
  force_new = FALSE,
  use_wd = TRUE,
  use_repos = FALSE,
  verbose = TRUE
)
```

Arguments

- **refs**: Reference for one or more GitHub repository in owner/repo format (e.g. "neurogenomics/rworkflows"), or an R package name (e.g. "rworkflows").
- **paths**: Paths to `DESCRIPTION` file(s) R package(s).
- **db**: A `data.table` of R package metadata generated by `biocPkgList`.
- **cache_dir**: Directory where to cache downloaded files.
- **force_new**: Ignore cached files and re-download them instead.
- **use_wd**: Search the local working directory (and the one above it) for `DESCRIPTION` files.
- **use_repos**: Use R standard R package repositories like CRAN and Bioc to find `DESCRIPTION` files.
- **verbose**: Print messages.

Value

A named list of `packageDescription` objects.
get_hex

Examples

d <- get_description(refs="neurogenomics/rworkflows")

get_hex

Get hex

Description

Get the URL of a hex sticker for a given R package (if one exists).

Usage

get_hex(
  refs = NULL,
  paths = here::here("DESCRIPTION"),
  hex_path = "inst/hex/hex.png",
  branch = c("master", "main", "dev"),
  hex_height = 300,
  check_url = TRUE,
  add_html = TRUE,
  verbose = TRUE
)

Arguments

refs Reference for one or more GitHub repository in owner/repo format (e.g. "neurogenomics/rworkflows"), or an R package name (e.g. "rworkflows").
paths Paths to DESCRIPTION file(s) R package(s).
hex_path Path to hex sticker file.
branch Name of the GitHub repository branch to use.
hex_height Height of the hex sticker in pixels (when add_hex=TRUE).
check_url Check whether the URL actually exists.
add_html Wrap the URL in an html "img" tag and set its height with hex_height.
verbose Print messages.

Value

URL

Examples

hex_url <- get_hex(refs=c("neurogenomics/rworkflows",
                           "neurogenomics/echolocatoR"))
infer_biocviews  

Infer the best terms to fill the `biocViews` field in your `DESCRIPTION` file based on the code within your R package. By default, also includes any `biocViews` that are already present in the `DESCRIPTION` file. Please see the Bioconductor website for more details.

Usage

```r
infer_biocviews(
  pkgdir = here::here(),
  branch = c("Software", "AnnotationData", "ExperimentData")[1],
  type = c("recommended", "current", "remove"),
  keep_current = TRUE,
  include_branch = TRUE,
  biocviews = NULL,
  add_newlines = FALSE,
  verbose = TRUE
)
```

Arguments

- `pkgdir`  
  The path of the package Directory.

- `branch`  
  The branch which your package will belong to. It can be either 'Software', 'AnnotationData' or 'ExperimentData'.

- `type`  
  Which element of the `recommendBiocViews` results list to return. If a vector is supplied, only the first value will be used.

- `keep_current`  
  Keep any `biocViews` terms that are already included in the `DESCRIPTION` file.

- `include_branch`  
  Whether to include the branch argument as one of the returned `biocViews`.

- `biocviews`  
  User-supplied `biocViews` terms to include in addition to the automated recommendations.

- `add_newlines`  
  Prefix each package name with a newline character and two spaces. This is useful for formatting `DESCRIPTION` files.

- `verbose`  
  Print messages.

Value

A character vector of `biocviews`. 
Examples

```r
## Don't run simply bc biocViews::recommendBiocViews is unable
## to find the DESCRIPTION file when running examples.
## Not run:
biocviews <- infer_biocviews()
## End(Not run)
```

### infer_deps

**Infer dependencies**

**Description**

Infers the R packages that your R package depends on.

**Usage**

```r
infer_deps(
  path = here::here(),
  which = c("Imports", "Suggests"),
  imports_thresh = 2,
  imports = NULL,
  suggests = c("testthat", "rmarkdown", "markdown", "knitr", "remotes", "knitr", "covr"),
  errors = c("reported", "fatal", "ignored"),
  dev = FALSE,
  progress = TRUE,
  add_newlines = FALSE
)
```

**Arguments**

- **path**: The path to a (possibly multi-mode) R file, or a directory containing such files. By default, all files within the current working directory are checked, recursively.
- **which**: Which types of dependencies to return.
- **imports_thresh**: The minimum number of times that a package has to be called within your package to assign it as an Import. If is called less times than this threshold, it will instead be assigned as a Suggest, which means it will not be installed by default.
- **imports**: R packages that are exempt from the suggests_thresh rule and are instead automatically assigned as Imports.
- **suggests**: R packages that are exempt from the suggests_thresh rule and are instead automatically assigned as Suggests.
- **errors**: How should errors that occur during dependency enumeration be handled? See **Errors** for more details.
is_gha

is_gha

Boolean; include 'development' dependencies as well? That is, packages which may be required during development but are unlikely to be required during runtime for your project. By default, only runtime dependencies are returned.

progress

Boolean; report progress output while enumerating dependencies?

add_newlines

Prefix each package name with a newline character and two spaces. This is useful for formatting DESCRIPTION files.

Value

A character vector of R package names.

Examples

deps <- infer_deps()

Description

Tests whether a function is currently being run within a GitHub Actions workflow or not.

Usage

is_gha(var = "GITHUB_ACTION", verbose = TRUE)

Arguments

var

Environmental variable to check.

verbose

Print messages.

Value

Boolean

Source

GitHub Actions docs

Examples

is_gha()
Description

Create one or more badges showing the status of your R package. Uses the package `badger`.

Usage

```r
use_badges(
    ref = NULL,
    add_hex = TRUE,
    add_actions = "rworkflows",
    add_doi = NULL,
    add_lifecycle = FALSE,
    add_github_version = TRUE,
    add_commit = TRUE,
    add_code_size = TRUE,
    add_license = TRUE,
    add_authors = TRUE,
    addCodecov = TRUE,
    addCodecov_graphs = "icicle",
    addBiocRelease = FALSE,
    addBiocDownload_month = FALSE,
    addBiocDownload_total = FALSE,
    addBiocDownload_rank = FALSE,
    addCranRelease = FALSE,
    addCranChecks = FALSE,
    addCranDownload_month = FALSE,
    addCranDownload_total = FALSE,
    branch = "master",
    as_list = FALSE,
    sep = "\n",
    hex_height = 300,
    codecov_graph_width = 200,
    colors = list(github = "black",
                  bioc = "green",
                  cran = "black",
                  default = "blue",
                  lifecycle = NULL),
    verbose = TRUE
)
```

Arguments

- **ref**: Reference for a GitHub repository. If `NULL` (the default), the reference is determined by the URL field in the DESCRIPTION file.
- **add_hex**: Add a hex sticker. If `add_hex=TRUE`, will assume the sticker is located at the following relative path: "inst/hex/hex.png". If `add_hex` is a character string, this will instead be used as the relative hex path (e.g. "/images/mysticker.png").
add_actions  The name of one or more GitHub Actions to show the status for with `badge_github_actions` (e.g. `c("rworkflows","rworkflows_static")).

add_doi  Add the DOI of a given package or publication associated with the package using `badge_doi`. Must be provided as a character string, e.g.: "10.1111/2041-210X.12628"

add_lifecycle  Add package lifecycle stage. If not FALSE, must be a character string indicating one of the following valid lifecycle stage:
- "stable"
- "deprecated"
- "superseded"
- "experimental"

See lifecycle.r-lib.org for further details.

add_github_version  Add package version with `badge_github_version`.

add_commit  Add the last GitHub repo commit date with `badge_last_commit`.

add_code_size  Add code size with `badge_code_size`.

add_license  Add license info with `badge_license`.

add_authors  Add author names inferred from the DESCRIPTION file.

add_codecov  Add Codecov status with `badge_codecov`. See the Codecov site for more information about these badges.

add_codecov_graphs  Add Codecov graphs visualising results of code coverage tests. Options include:
- "sunburst"
- "tree"
- "icicle"

See the Codecov site for more information about each plot type.

add_bioc_release  Add Bioc release version with `badge_bioc_release`.

add_bioc_download_month  Add the number of Bioc downloads last month `badge_bioc_download`.

add_bioc_download_total  Add the number of Bioc downloads total `badge_bioc_download`.

add_bioc_download_rank  Add the download rank of the package on Bioc `badge_bioc_download_rank`.

add_cran_release  Add Bioc release version with `badge_cran_release`.

add_cran_checks  Add whether package is passing all checks on CRAN with `badge_cran_checks`.

add_cran_download_month  Add the number of CRAN downloads last month `badge_cran_download`.

add_cran_download_total  Add the number of CRAN downloads total `badge_cran_download`.

branch  Name of the GitHub repository branch to use.
use_dockerfile

as_list: Return the header as a named list (TRUE), or a collapsed text string (default: FALSE).
sep: Character to separate each item in the list with using `paste`
hex_height: Height of the hex sticker in pixels (when `add_hex=TRUE`).
codecov_graph_width: Width of each Codecov graph in pixels (when `add_codecov_graph!=FALSE`).
colors: Colors to assign to each group of badges (when possible).
verbose: Print messages.

Value
A named list of selected badges in markdown format.

Examples

```r
badges <- rworkflows::use_badges(ref = "neurogenomics/rworkflows")
```

---

use_dockerfile | Use Dockerfile

Description

Creates a Docker file to be used with the GitHub Actions (GHA) workflows distributed by `rworkflows`.

Usage

```r
use_dockerfile(
  save_dir = here::here(),
  path = file.path(save_dir, "Dockerfile"),
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

Arguments

- `save_dir`: Directory to save the Docker file to.
- `path`: Path to the Docker file.
- `force_new`: If a Docker file already exists, overwrite it (default: FALSE).
- `show`: Print the contents of the Docker file in the R console.
- `verbose`: Print messages.

Value

Path to Docker file.
**use_issue_template**

**Examples**

```r
path <- use_dockerfile(save_dir=tempdir())
```

---

**Description**

Creates one or more Issue Templates to be used in a GitHub repository.

**Usage**

```r
use_issue_template(
  templates = c("bug_report.md", "feature_request.md"),
  save_dir = here::here(".github", "ISSUE_TEMPLATE"),
  path = file.path(save_dir, templates),
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

**Arguments**

- `templates`: The names of templates to be used.
- `save_dir`: Directory to save the Docker file to.
- `path`: Path to the Docker file.
- `force_new`: If a Docker file already exists, overwrite it (default: `FALSE`).
- `show`: Print the contents of the Docker file in the R console.
- `verbose`: Print messages.

**Value**

Path to Issue Templates.

**Examples**

```r
path <- use_issue_template(save_dir=tempdir())
```
use_readme  

Use README

Description

Creates an rmarkdown README file that autofills using metadata from the R package DESCRIPTION file.

Usage

```r
use_readme(
  save_dir = here::here(),
  path = file.path(save_dir, "README.Rmd"),
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

Arguments

- `save_dir`: Directory to save the vignette file to.
- `path`: Path to the vignette file.
- `force_new`: If the file already exists, overwrite it (default: FALSE).
- `show`: Print the contents of the vignette file in the R console.
- `verbose`: Print messages.

Value

Path to README file.

Examples

```r
## use default save_dir in practice
path <- use_readme(save_dir = tempdir())
```

---

use_vignette_docker  

Use vignette: Docker

Description

Creates a vignette rmarkdown file demonstrates how to create a Docker/Singularity image from a container stored in Dockerhub.
Usage

```r
use_vignette_docker(
  docker_org,
  title = "Docker/Singularity Containers",
  vignette_index_entry = "docker",
  save_dir = here::here(),
  path = file.path(save_dir, "vignettes", "docker.Rmd"),
  output = "BiocStyle::html_document",
  port_in = 8787,
  port_out = 8900,
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

Arguments

- **docker_org**: DockerHub organization name. Can simply be your Dockerhub username instead.
- **title**: Title of vignette.
- **vignette_index_entry**: Index entry of the vignette, which is used when creating the navigation bar in the *pkgdown* site.
- **save_dir**: Directory to save the vignette file to.
- **path**: Path to the vignette file.
- **output**: Vignette output style. Defaults to *html_document*.
- **port_in**: Port number to route into the docker container. See the Docker docs for further details.
- **port_out**: Port number to route out of docker container. See the Docker docs for further details.
- **force_new**: If the file already exists, overwrite it (default: *FALSE*).
- **show**: Print the contents of the vignette file in the R console.
- **verbose**: Print messages.

Value

Path to vignette file.

Examples

```r
path <- use_vignette_docker(docker_org = "neurogenomicslab",
                           ## use default save_dir in practice
                           save_dir = tempdir())
```
Description

Creates a "Get started" rmarkdown vignette file.

Usage

use_vignette_getstarted(
    package,
    title = "Get started",
    vignette_index_entry = package,
    save_dir = here::here(),
    path = file.path(save_dir, "vignettes", paste0(package, ".Rmd")),
    output = "BiocStyle::html_document",
    force_new = FALSE,
    show = FALSE,
    verbose = TRUE
)

Arguments

package R package name.
title Title of vignette.
vignette_index_entry

Index entry of the vignette, which is used when creating the navigation bar in
the pkgdown site.

save_dir Directory to save the vignette file to.
path Path to the vignette file.
output Vignette output style. Defaults to html_document.
force_new If the file already exists, overwrite it (default: FALSE).
show Print the contents of the vignette file in the R console.
verbose Print messages.

Value

Path to vignette file.

Examples

path <- use_vignette_getstarted(package = "mypackage",
    ## use default save_dir in practice
    save_dir = tempdir())
use_workflow

Use GitHub Actions workflow

Description

Create workflow that calls an rworkflows GitHub Actions (GHA)

Usage

```r
use_workflow(
  template = "rworkflows",
  name = template,
  tag = "@master",
  on = c("push", "pull_request"),
  branches = c("master", "main", "devel", "RELEASE_*"),
  runners = construct_runners(),
  run_bioccheck = FALSE,
  run_rcmdcheck = TRUE,
  as_cran = TRUE,
  run_vignettes = TRUE,
  has_testthat = TRUE,
  run_covr = TRUE,
  run_pkgdown = TRUE,
  has_runit = FALSE,
  has_latex = FALSE,
  tinytex_installer = "TinyTeX-1",
  tinytex_version = ",
  pandoc_version = "2.19",
  run_docker = FALSE,
  github_token = "$\{\text{secrets.GITHUB_TOKEN}\}$",
  docker_user = NULL,
  docker_org = docker_user,
  docker_token = "$\{\text{secrets.DOCKER_TOKEN}\}$",
  cache_version = "cache-v1",
  enable_act = FALSE,
  save_dir = here::here(".github", "workflows"),
  return_path = TRUE,
  force_new = FALSE,
  preview = FALSE,
  verbose = TRUE
)
```

Arguments

- **template**: Workflow template name.
use_workflow

• "rworkflows" A short workflow script that calls the GitHub action from the GitHub Marketplace. The action is continually updated so users do not need to worry about maintaining it.
• "rworkflows_static" A longer workflow script that explicitly copies all steps from the `rworkflows` action into a static file. Users may need to update this file themselves over time, though this does allow for a fully customisable workflow.

name
An arbitrary name to call the workflow.
tag
Which version of the `rworkflows` action to use. Can be a branch name on the GitHub repository (e.g. "@master"), or a Release Tag (e.g. "@v1").
on
GitHub trigger conditions.
branch
GitHub trigger branches.
runtimes
Runner configurations for multiple Operating Systems (OS), including R versions, Bioc versions, and container sources. Can use the `construct_runners` functions to assist in constructing customized runners configurations.
run_bioccheck
Run Bioconductor checks using `BiocCheck::BiocCheck()`. Must pass in order to continue workflow.
run_rcmdcheck
Run R CMD checks using `rcmdcheck::rcmdcheck()`. Must pass in order to continue workflow.
as_cran
When running R CMD checks, use the `as-cran` flag to apply CRAN standards
run_vignettes
Build and check R package vignettes.
has_testthat
Run unit tests and report results.
run_covr
Run code coverage tests and publish results to codecov.
run_pkgdown
Knit the `README.Rmd` (if available), build documentation website, and deploy to `gh-pages` branch.
has_runit
Run R Unit tests.
has_latex
Install a suite of LaTeX dependencies used for rendering Sweave (.rnw) and other documentation files.
tinytex_installer
Which release of tinytex (bundles of LaTeX packages) to use. All options can be found here. Note, 'TinyTeX-2' is only available for tinytex_version='daily'.
tinytex_version
Which version of tinytex to use. When set to '', uses the latest daily build. All versions can be found here.
pandoc_version
Which version of pandoc to use. For details see here.
run_docker
Whether to build and push a Docker container to DockerHub.
github_token
GitHub authentication token with permissions to push to the R package’s GitHub repository. Also used to bypass GitHub download limits. By default, uses `{{ secrets.GITHUB_TOKEN }}` which is automatically set up by GitHub. However users can also choose to pass a custom GitHub secret variable (e.g. `{{ secrets.PAT_GITHUB }}`) which allows access to private repositories. Read here for more details.
docker_user
DockerHub username.
use_workflow

docker_org  DockerHub organization name. Is the same as docker_user by default.
docker_token DockerHub token.
cache_version Name of the cache subdirectory to be used when reinstalling software in GHA.
enable_act  Whether to add extra lines to the yaml to enable local workflow checking with act.
save_dir  Directory to save workflow to.
return_path Return the path to the saved yaml workflow file (default: TRUE), or return the yaml object directly.
force_new  If the GHA workflow yaml already exists, overwrite with new one (default: FALSE).
preview  Print the yaml file to the R console.
verbose  Print messages.

Value
Path or yaml object.

Source
Issue reading in "on:'y','n' elements.
Issue writing "on:' as "as':" 

Examples
path <- use_workflow(save_dir = file.path(tempdir(),".github","workflows"))
Index

* datasets
  - biocpkgtools_db, 2
    - badge_bioc_download, 16
    - badge_bioc_download_rank, 16
    - badge_bioc_release, 16
    - badge_code_size, 16
    - badge_codecov, 16
    - badge_cran_checks, 16
    - badge_cran_download, 16
    - badge_cran_release, 16
    - badge_doi, 16
    - badge_github_actions, 16
    - badge_github_version, 16
    - badge_last_commit, 16
    - badge_license, 16
    - bioc_r_versions, 3, 6
    - biocPkgList, 2, 10
    - biocpkgtools_db, 2
  - check_cont, 5, 6
  - construct_authors, 4, 4, 8
  - construct_cont, 4
  - construct_runners, 5, 23
  - data.table, 7, 10
  - desc, 7
  - description, 9
  - dt_to_desc, 7
  - fill_description, 7
  - get_description, 10
  - get_hex, 11
  - html_document, 20, 21
  - infer_biocviews, 9, 12
  - infer_deps, 13
  - is_gha, 14
  - paste, 17
  - person, 4, 8
  - recommendBiocViews, 12
  - use_badges, 15
  - use_dockerfile, 17
  - use_issue_template, 18
  - use_readme, 19
  - use_vignette_docker, 19
  - use_vignette_getstarted, 21
  - use_workflow, 22