

# Package: changer (via r-universe)

August 27, 2024

**Type** Package

**Title** Change R Package Name

**Version** 0.0.5

**Date** 2022-05-03

**Description** Changing the name of an existing R package is annoying but common task especially in the early stages of package development. This package (mostly) automates this task.

**License** GPL (>= 2)

**Imports** available, devtools, git2r

**Depends** R (>= 3.5.0)

**Encoding** UTF-8

**RoxygenNote** 7.1.2

**URL** <https://github.com/helske/changer>

**BugReports** <https://github.com/helske/changer/issues>

**Repository** <https://helske.r-universe.dev>

**RemoteUrl** <https://github.com/helske/changer>

**RemoteRef** HEAD

**RemoteSha** ce15565a4c01a75d1b5dff962ad619441dae1b1b

## Contents

changer . . . . .	2
<b>Index</b>	<b>4</b>

---

`changer`*Change Name of Existing R Package*

---

## Description

Changing the name of an existing R package is annoying but common task especially in the early stages of package development. Function `changer` tries to automate this task. See README for more information.

## Usage

```
changer(  
  path,  
  new_name,  
  check_validity = TRUE,  
  change_git = TRUE,  
  run_roxygen = FALSE,  
  remote_name = NULL,  
  ask = TRUE,  
  replace_all = FALSE  
)
```

## Arguments

<code>path</code>	Path of the package.
<code>new_name</code>	Desired name of the package.
<code>check_validity</code>	Check first if the package name is valid and available by running the function <code>available</code> from the package <code>available</code> . Default is <code>TRUE</code> . This will prompt a warning about potentially offensive results from Urban Dictionary which is used to check the validity of the package name.
<code>change_git</code>	If <code>TRUE</code> (default), changes the remote url of the remote. Note that you still need to change the name of the GitHub repository manually as follows: Go to the URL of your Github package, click Settings, change the name under "Repository name", and click Rename.
<code>run_roxygen</code>	Should the package documentation be updated via roxygen? If <code>TRUE</code> , removes all old Rd files in <code>man</code> directory.
<code>remote_name</code>	Name of the remote. Defaults to <code>git2r::remotes(path)[1]</code> .
<code>ask</code>	Ask confirmation before starting the rename process. Default is <code>TRUE</code> .
<code>replace_all</code>	If <code>TRUE</code> , replace all occurrences of the old package name in files, also in parts of the word. Default is <code>FALSE</code> , so only whole words are replaced.

## Details

Note that if the package is already published in CRAN, then the name change is more problematic (you need to ask CRAN for permission first).

If the package is also available in Github, then you need to do the following:

Go to the URL of your Github package, click Settings near the top-right, change the name under "Repository name", and finally click Rename.

Warnings:

If the current name of your package is just some commonly used word (such as "package"), then you are in trouble, as find and replace will change all of those words to new\_name as well.

If you have a function with same name as your package, that will change as well.

It is strongly recommended to make backups before proceeding.

Inspired by Nick Tierney's blog post: <https://www.njtierney.com/post/2017/10/27/change-pkg-name/>

## Examples

```
content <- letters
package.skeleton("package.with.boring.name", path = tempdir())
readLines(file.path(tempdir(), "package.with.boring.name", "DESCRIPTION"))

changer(file.path(tempdir(), "package.with.boring.name"),
  new_name = "superpack", check_validity = FALSE, ask = FALSE)
readLines(file.path(tempdir(), "superpack", "DESCRIPTION"))
unlink(file.path(tempdir(), "superpack"), recursive = TRUE)
```

# Index

changer, [2](#)