

# Package ‘default’

October 13, 2022

**Type** Package

**Title** Change the Default Arguments in R Functions

**Date** 2017-08-07

**Version** 1.0.0

**Author** Nick Golding

**Maintainer** Nick Golding <nick.golding.research@gmail.com>

**Description** A simple syntax to change the default values for function arguments, whether they are in packages or defined locally.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Suggests** testthat

**LazyData** true

**RoxygenNote** 6.0.1

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2017-08-07 05:54:57 UTC

## R topics documented:

default . . . . . 1

**Index** 3

---

default                    *change a function’s default arguments*

---

## Description

default() lets you check, and change, a function’s default arguments. reset\_default() returns the arguments to their original defaults.

**Usage**

```
default(fun)

default(fun) <- value

reset_default(fun)
```

**Arguments**

|       |   |
|-------|---|
| fun   | a function  |
| value | a named list of new default arguments for that function |

**Details**

If `fun` is a function from a package, a function of the same name will be defined in the calling environment (e.g. your workspace). If `fun` is defined locally, it will be overwritten by the version with the new defaults.

`reset_default` *returns* the reset function, rather than modifying it in place, so you'll need to reassign it, as in the example.

**Value**

`default()` (without assignment) invisibly returns a pairlist of the current values of the default arguments. It also prints the default arguments, highlighting those that the user has changed from their original defaults.

`reset_default()` returns the `fun`, but with the defaults reset to their original values. If `fun` was a function from a package, the same thing can be achieved by replacing the locally-defined version of the function.

**Examples**

```
# list the default arguments for a function
default(data.frame)

# change one or more of them
default(data.frame) <- list(fix.empty.names = FALSE)
data.frame(1:3)

# reset the defaults
data.frame <- reset_default(data.frame)
data.frame(1:3)
```

# Index

`default`, [1](#)

`default<- (default)`, [1](#)

`reset_default (default)`, [1](#)