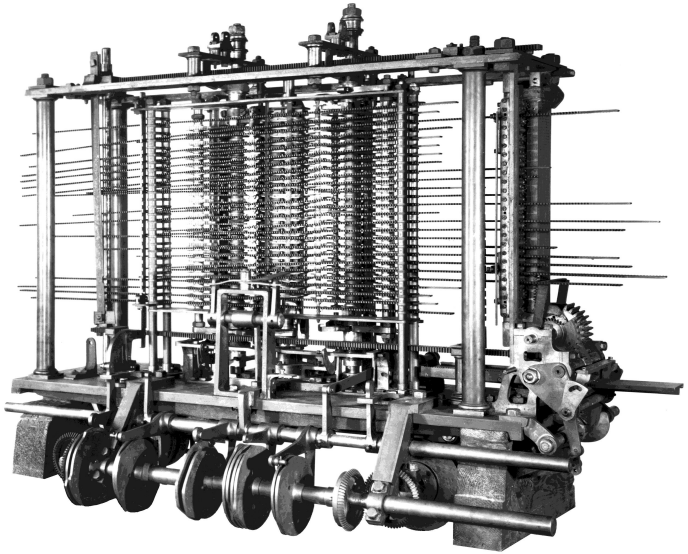
LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Advanced Programming in R

Binder, Schalk



Unit: R Package

What is an R Package?

The "memoise" package

'Memoisation' of Functions

Documentation for package ‘memoise’ version 2.0.1

- DESCRIPTION file.

Help Pages

cache_filesystem	Filesystem Cache
cache_gcs	Google Cloud Storage Cache Google Cloud Storage backed cache, for remote caching.
cache_memory	In Memory Cache
cache_s3	Amazon Web Services S3 Cache Amazon Web Services S3 backed cache, for remote caching.
drop_cache	Drops the cache of a memoised function for particular arguments.
forget	Forget past results. Resets the cache of a memoised function. Use 'drop_cache' to reset the cache only for particular arguments.
has_cache	Test whether a memoised function has been cached for particular arguments.
is.memoised	Test whether a function is a memoised copy. Memoised copies of functions carry an attribute 'memoised = TRUE', which is what 'is.memoised()' tests for.
is.memoized	Test whether a function is a memoised copy. Memoised copies of functions carry an attribute 'memoised = TRUE', which is what 'is.memoised()' tests for.
memoise	Memoise a function
memoize	Memoise a function
timeout	Return a new number after a given number of seconds

Showing this package because it is relatively small and simple.

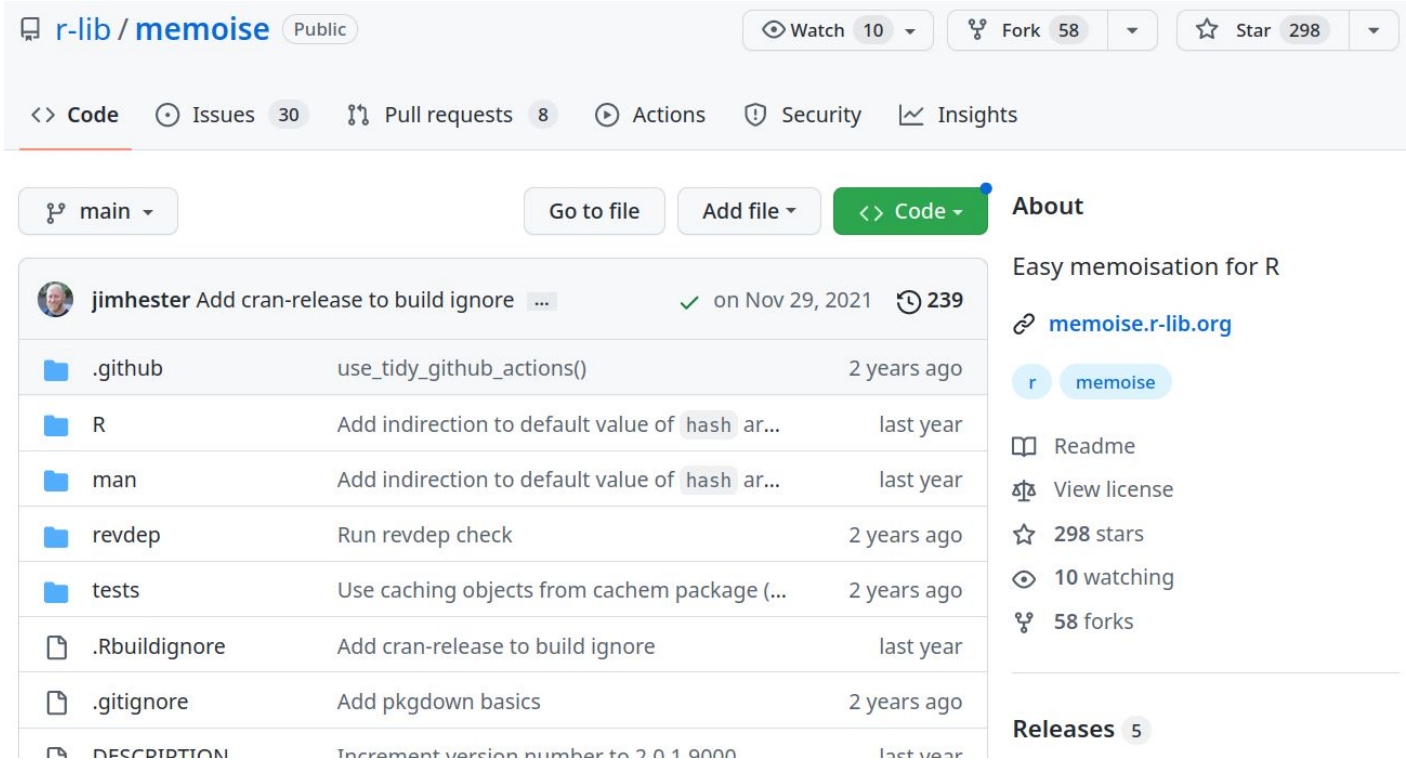
The "memoise" package

`memoise: 'Memoisation' of Functions`

Cache the results of a function so that when you call it again with the same arguments it returns the previously computed value.

Version:	2.0.1
Imports:	rlang (≥ 0.4.10), cachem
Suggests:	digest, aws.s3, covr, googleAuthR, googleCloudStorageR, httr, testthat
Published:	2021-11-26
Author:	Hadley Wickham [aut], Jim Hester [aut], Winston Chang [aut, cre], Kirill Müller [aut], Daniel Cook [aut], Mark Edmondson [ctb]
Maintainer:	Winston Chang <winston at rstudio.com>
BugReports:	https://github.com/r-lib/memoise/issues
License:	MIT + file LICENSE
URL:	https://memoise.r-lib.org, https://github.com/r-lib/memoise
NeedsCompilation:	no

The "memoise" package



The "memoise" package

README.md

memoise

CRAN status 

The memoise package makes it easy to memoise R functions. **Memoisation** (https://en.wikipedia.org/wiki/Memoization) caches function calls so that if a previously seen set of inputs is seen, it can return the previously computed output.

Installation

Install from CRAN with

```
install.packages("memoise")
```

Usage

To memoise a function, use `memoise()`:

```
library(memoise)
f <- function(x) {
  Sys.sleep(1)
  mean(x)
}
mf <- memoise(f)
```

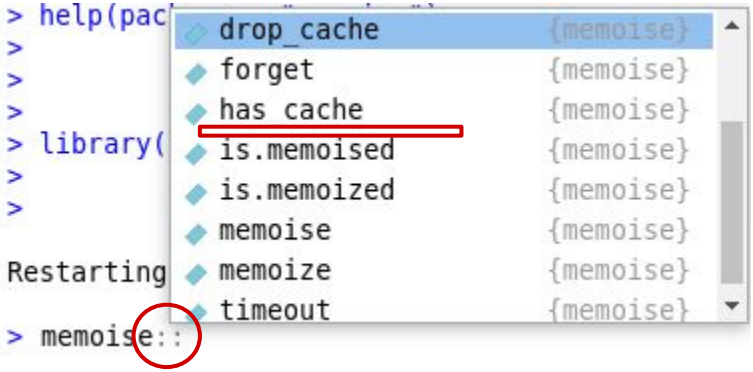
What happens here?

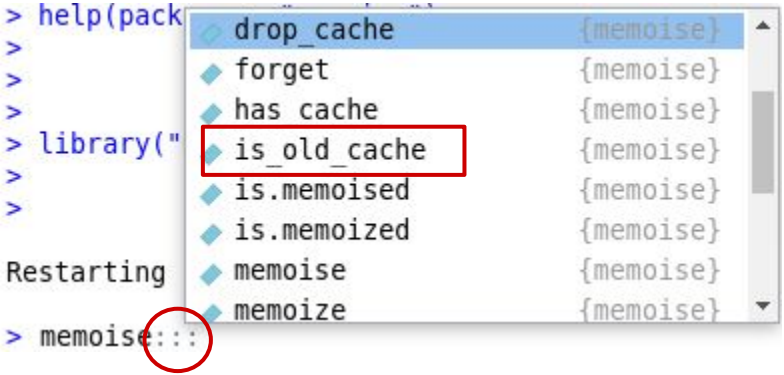
```
> install.packages("memoise")
Installing package into '/home/user/R/x86_64-redhat-linux-gnu-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/src/contrib/memoise_2.0.1.tar.gz'
Content type 'application/x-gzip' length 17852 bytes (17 KB)
==================================================
downloaded 17 KB

* installing *source* package 'memoise' ...
** package 'memoise' successfully unpacked and MD5 sums checked
** using staged installation
** R
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
  converting help for package 'memoise'
    finding HTML links ... done
    cache_filesystem                        html
    cache_gcs                               html
    cache_memory                            html
    cache_s3                                html
    drop_cache                              html
    forget                                  html
    has_cache                               html
    is.memoised                             html
    memoise                                 html
    timeout                                 html
** building package indices
** testing if installed package can be loaded from temporary location
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (memoise)

The downloaded source packages are in
        '/tmp/RtmpSKonrn/downloaded packages'
```

download: **.tar.gz**-file

installation:
- check that the file is complete and correct
- do some preparation for faster loading
- check that it actually works with the system

What is in a package? -- exported objects





The "memoise" package -- exported objects

```
> memoise::forget
function (f)
{
    if (!is.memoised(f)) {
        return(FALSE)
    }
    env <- environment(f)
    if (!exists("_cache", env, inherits = FALSE))
        return(FALSE)
    cache <- get("_cache", env)
    cache$reset()
    TRUE
}
<bytecode: 0x55d7456bffb8>
<environment: namespace:memoise>
```

```
> forget
Error: object 'forget' not found
Environment:
  1:
  0:

> library("memoise")
> forget
function (f)
{
    if (!is.memoised(f)) {
        return(FALSE)
    }
    env <- environment(f)
        .
        .
        .
```

The "memoise" package -- exported objects

```
> search()
 [1] ".GlobalEnv"        "tools:rstudio"     "package:stats"
 [4] "package:graphics"  "package:grDevices" "package:utils"
 [7] "package:datasets"  "package:methods"   "Autoloads"
[10] "package:base"
> library("memoise")
> search()
 [1] ".GlobalEnv"        "package:memoise"   "tools:rstudio"
 [4] "package:stats"     "package:graphics"  "package:grDevices"
 [7] "package:utils"     "package:datasets"  "package:methods"
[10] "Autoloads"         "package:base"

> .GlobalEnv
<environment: R_GlobalEnv>
> parent.env(.GlobalEnv)
<environment: package:memoise>
attr(,"name")
[1] "package:memoise"
attr(,"path")
[1] "/home/user/R/x86_64-redhat-linux-gnu-library/4.1/memoise"
> parent.env(parent.env(.GlobalEnv))
<environment: 0x5601c1062750>
attr(,"name")
[1] "tools:rstudio"
```

Loading a package via library() adds the corresponding "package" environment to the chain of parent environments of .GlobalEnv.

When we access "forget", R searches inside .GlobalEnv (doesn't find it), then in .GlobalEnv's parent (where it is present)

The "memoise" package -- exported objects

```
> library("memoise")
> is_old_cache
Error: object 'is_old_cache' not found
Environment:
  1:
  0:

> memoise::is_old_cache
Error: 'is_old_cache' is not an exported object from 'namespace:memoise'
Environment:
  1:
  0:

> memoise:::is_old_cache
function (x)
{
    is.function(x$digest) && is.function(x$set) && is.function(x$get) &&
        is.function(x$has_key)
}
<bytecode: 0x55d748a6cf98>
<environment: namespace:memoise>
```

The "memoise" package -- exported objects

```
> memoise::forget
function (f)
{
    if (!is.memoised(f)) {
        return(FALSE)
    }
    env <- environment(f)
    if (!exists("_cache", env, inherits = FALSE))
        return(FALSE)
    cache <- get("_cache", env)
    cache$reset()
    TRUE
}
<bytecode: 0x55d7456bffb8>
<environment: namespace:memoise>
```

```
> parent.env(.GlobalEnv)
<environment: package:memoise>
attr(,"name")
[1] "package:memoise"
attr(,"path")
[1] "/home/user/R/x86_64-redhat-linux-gnu-library/4.1/memoise"
```

Notice the difference!
"package" vs. "namespace"

The "memoise" package -- exported objects

```
> names(parent.env(.GlobalEnv))
 [1] "cache_s3"          "forget"            "cache_filesystem"
 [4] "cache_gcs"         "timeout"           "is.memoised"
 [7] "memoize"           "memoise"           "has_cache"
[10] "drop_cache"        "cache_memory"      "is.memoized"
> names(environment(forget))
 [1] "forget"                "cache_s3"              "cache_gcs"
 [4] "cache_filesystem"      "timeout"               "memoize"
 [7] "is.memoised"           "validate_formulas"     "memoise"
[10] "wrap_old_cache"        "has_cache"             ".packageName"
[13] "drop_cache"            "cache_memory"          "is_old_cache"
[16] "is.memoized"           ".__NAMESPACE__."       ".__S3MethodsTable__."
[19] "print.memoised"
```

The "package": what the user gets access to

All of this, plus internal functions, plus some R internals

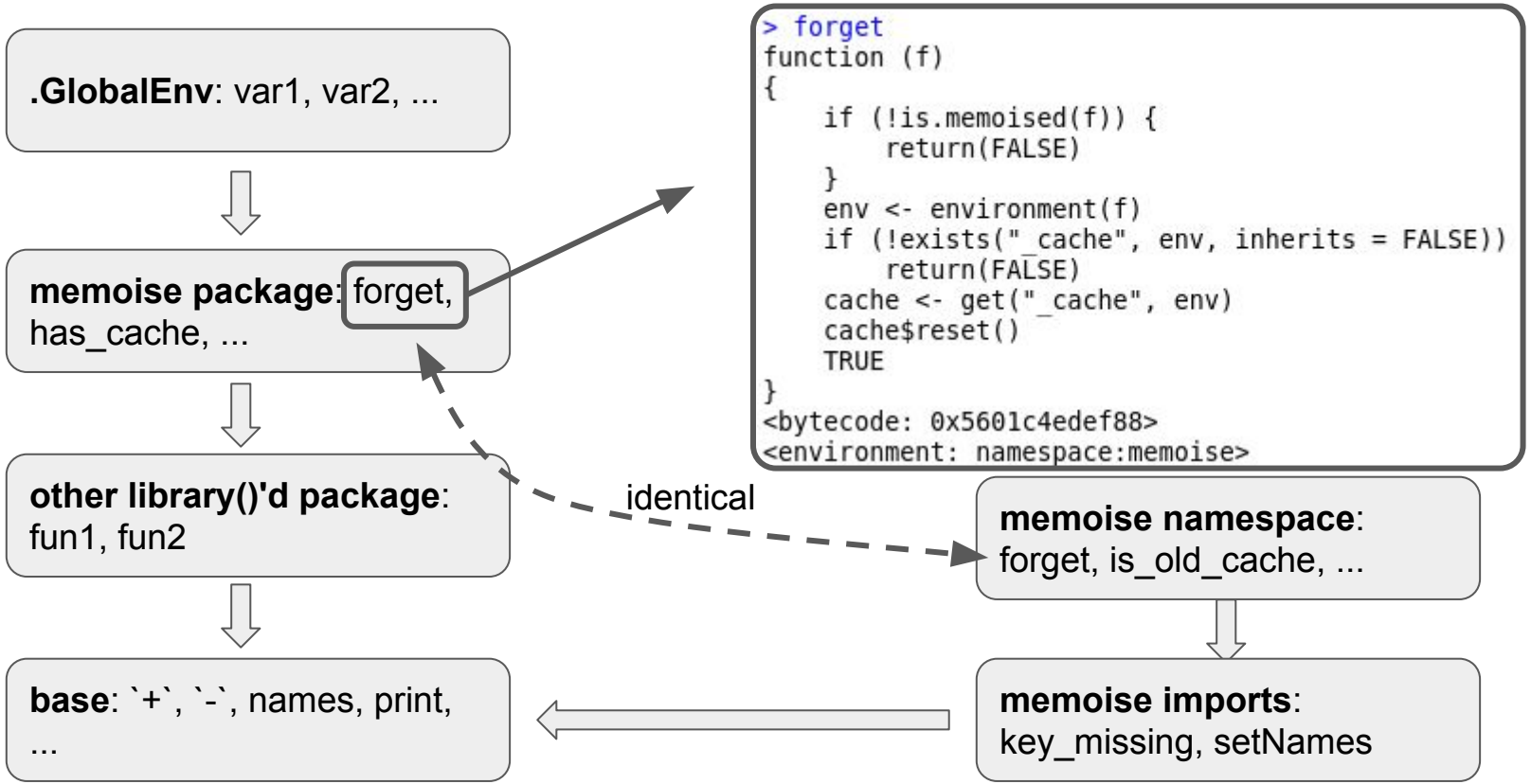
```
> parent.env(environment(forget))
<environment: 0x5601c499a480>
attr(,"name")
[1] "imports:memoise"
> names(parent.env(environment(forget)))
[1] "key missing" "setNames"
```

Parent of the "namespace" is an environment of "imports"

```
> parent.env(parent.env(environment(forget)))
<environment: namespace:base>
```

... and the parent of *that* is **base**. this is *not* .GlobalEnv! The package does not get access to user variables / packages.

The "memoise" package -- exported objects

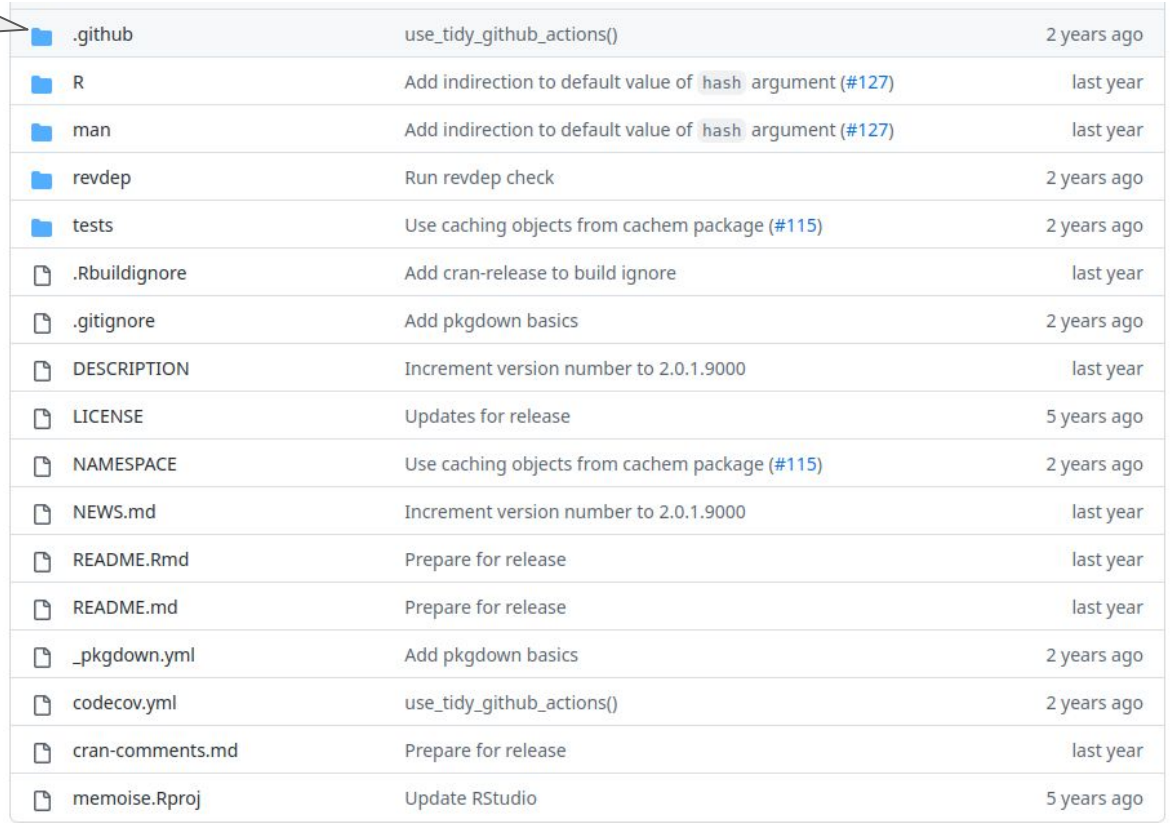


R Package from the Inside

Components of an R Package

Continuous Integration

Contains a script that is run by GitHub, e.g. to run automatic tests, possibly also to create a website with documentation

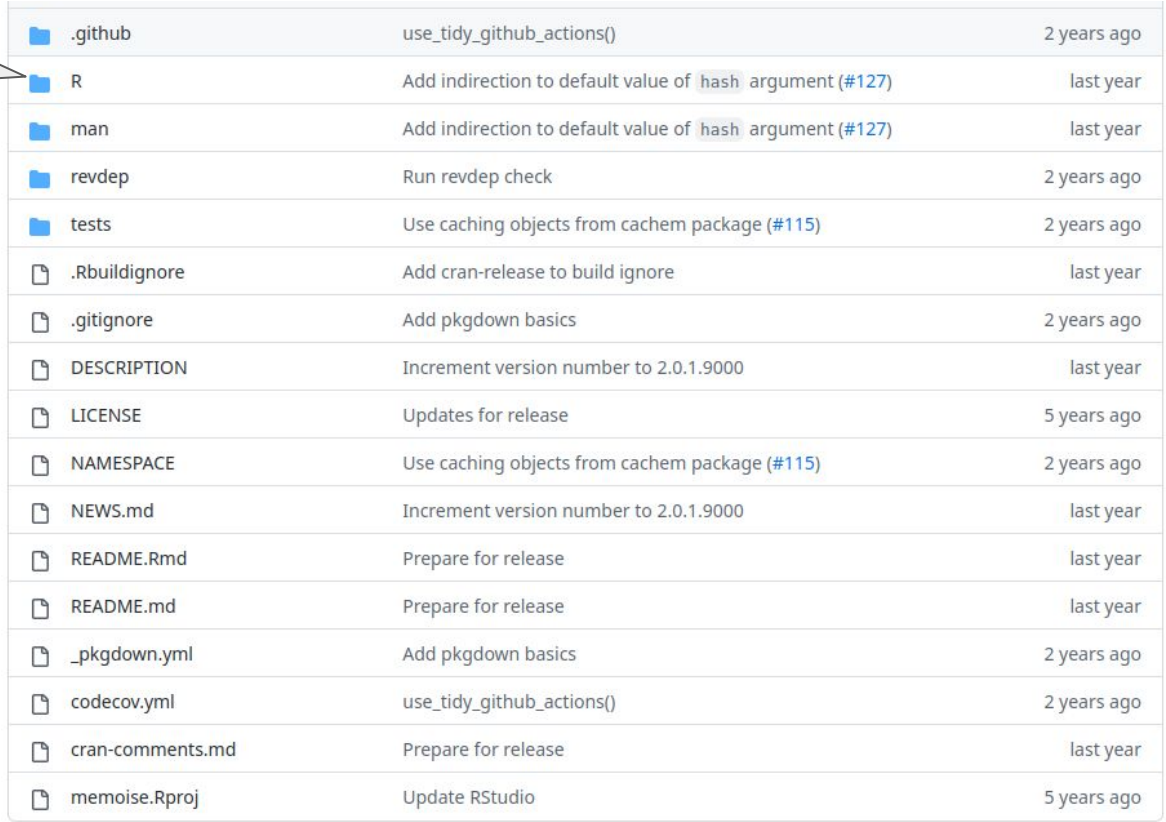


.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

R-Code

The code of the functions of the package, both the exported ones and the internal ones.

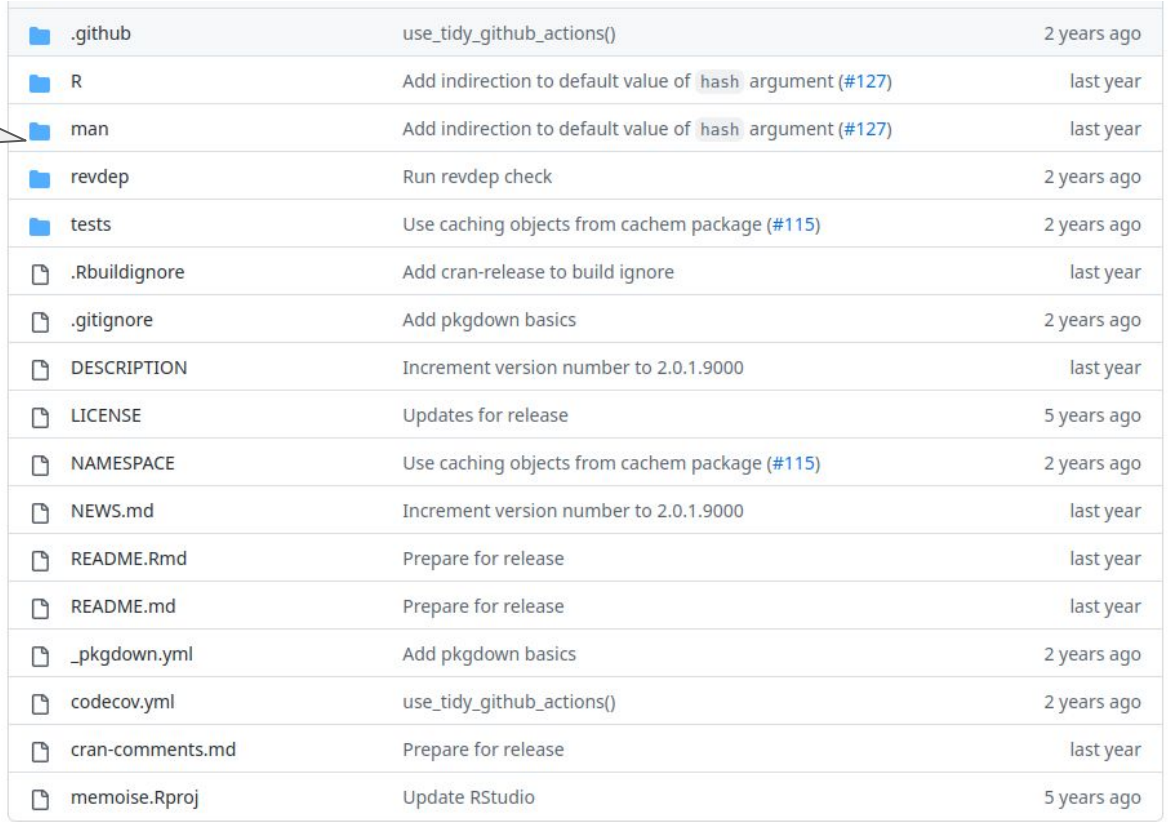


.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

Documentation.

The documentation is written in some LaTeX-like format. However, most people auto-generate this folder from comments in their code, usingthe **roxygen** package.



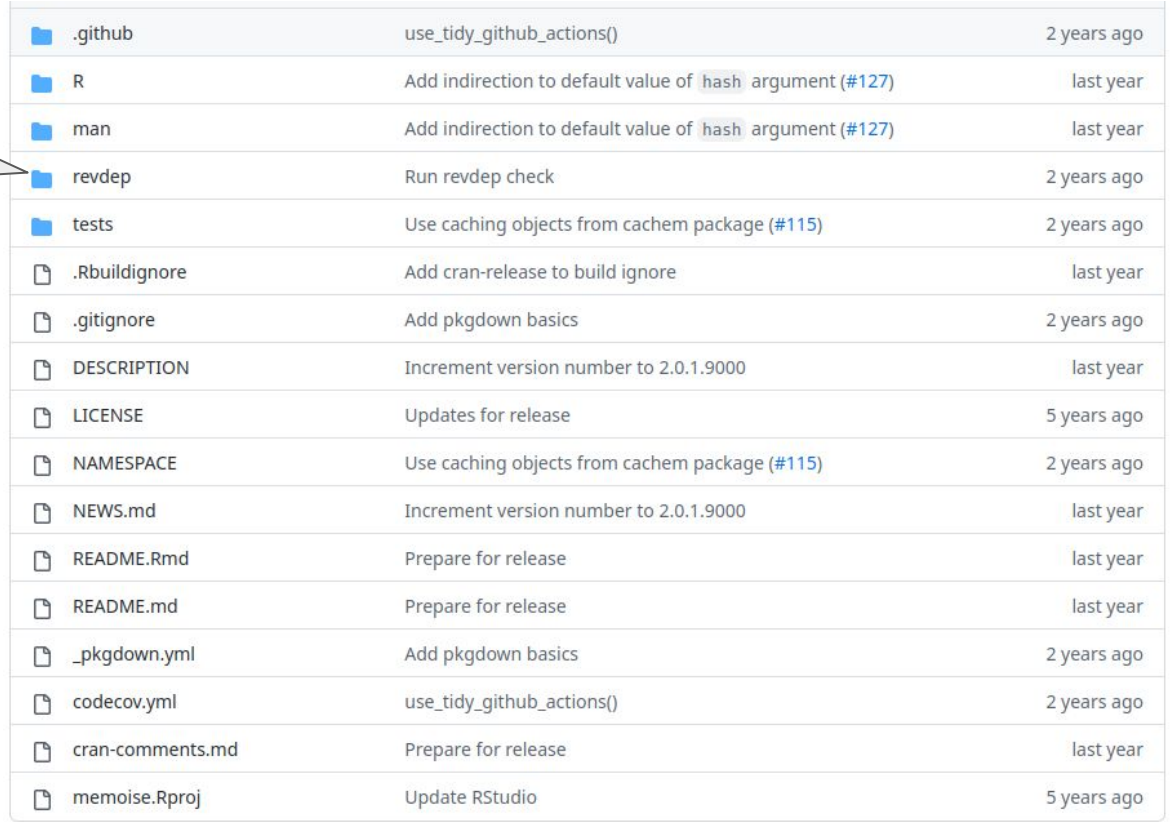
.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

Whatever you want.

R package repos can contain anything else that you want, e.g. some experimental scripts that you use, a paper you have written, images for the README.md file etc.

To make R ignore these folders, it is necessary to list them in the **.Rbuildignore** file.



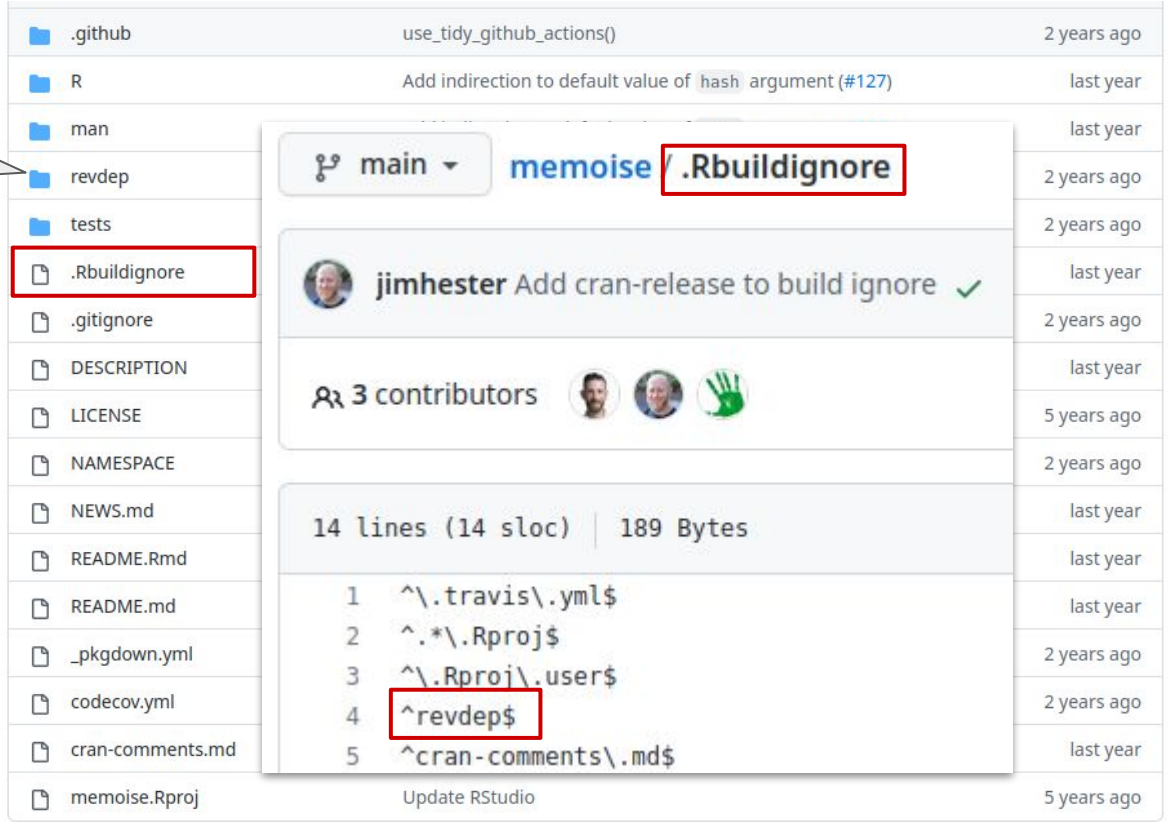
.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

Whatever you want.

R package repos can contain anything else that you want, e.g. some experimental scripts that you use, a paper you have written, images for the README.md file etc.

To make R ignore these folders, it is necessary to list them in the **.Rbuildignore** file.



Components of an R Package

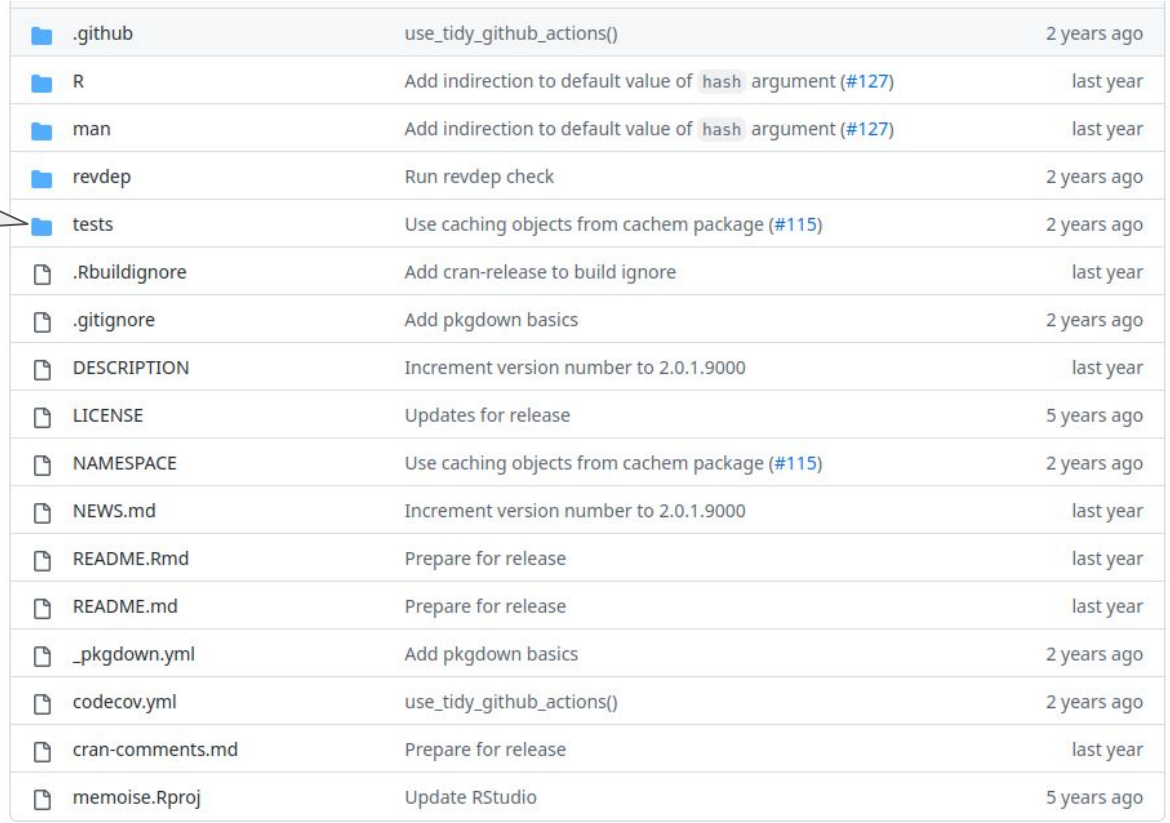
Tests.

Tests are an important aspect of software development.

Tests here have **2** purposes:

(1) They are run by the package maintainer (usually using CI) to check that new code does not break the package

(2) They are run by CRAN to check that changes in other libraries or in R itself do not break the package.

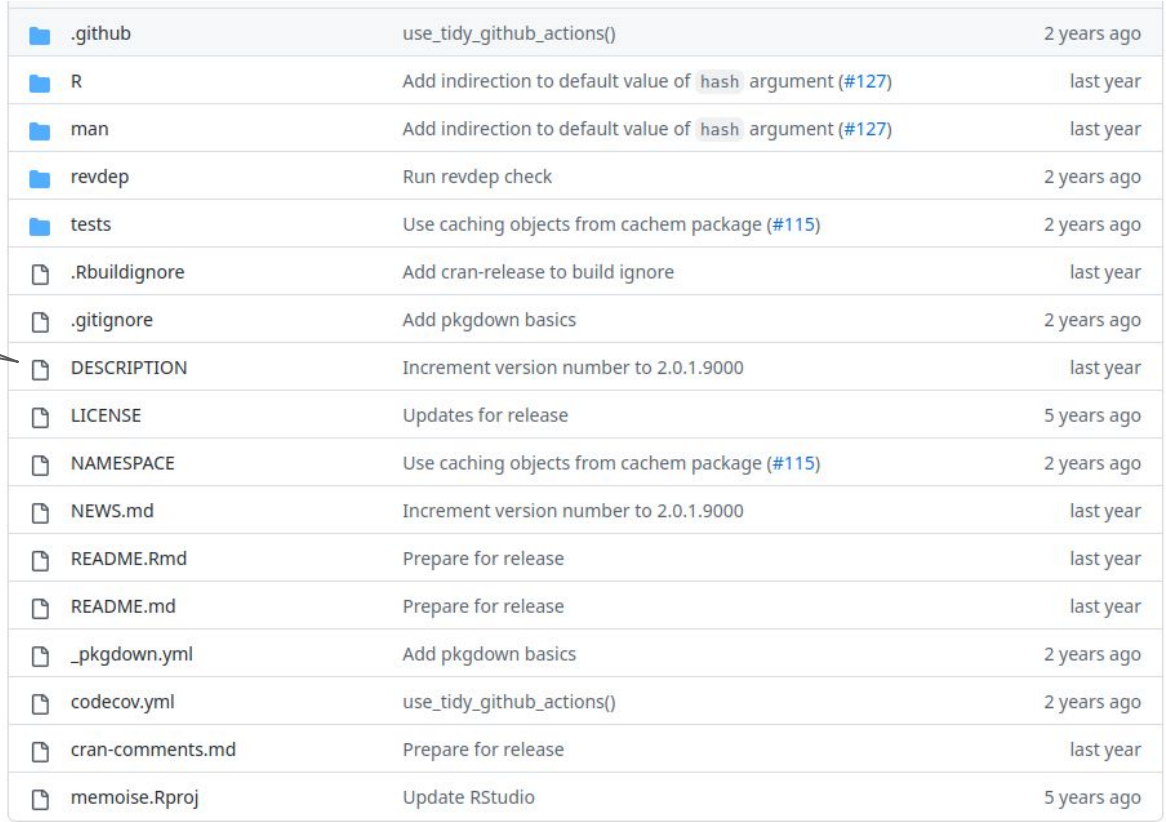


.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

DESCRIPTION

Meta-info about the package: Name, name of the author, license, package dependencies etc.

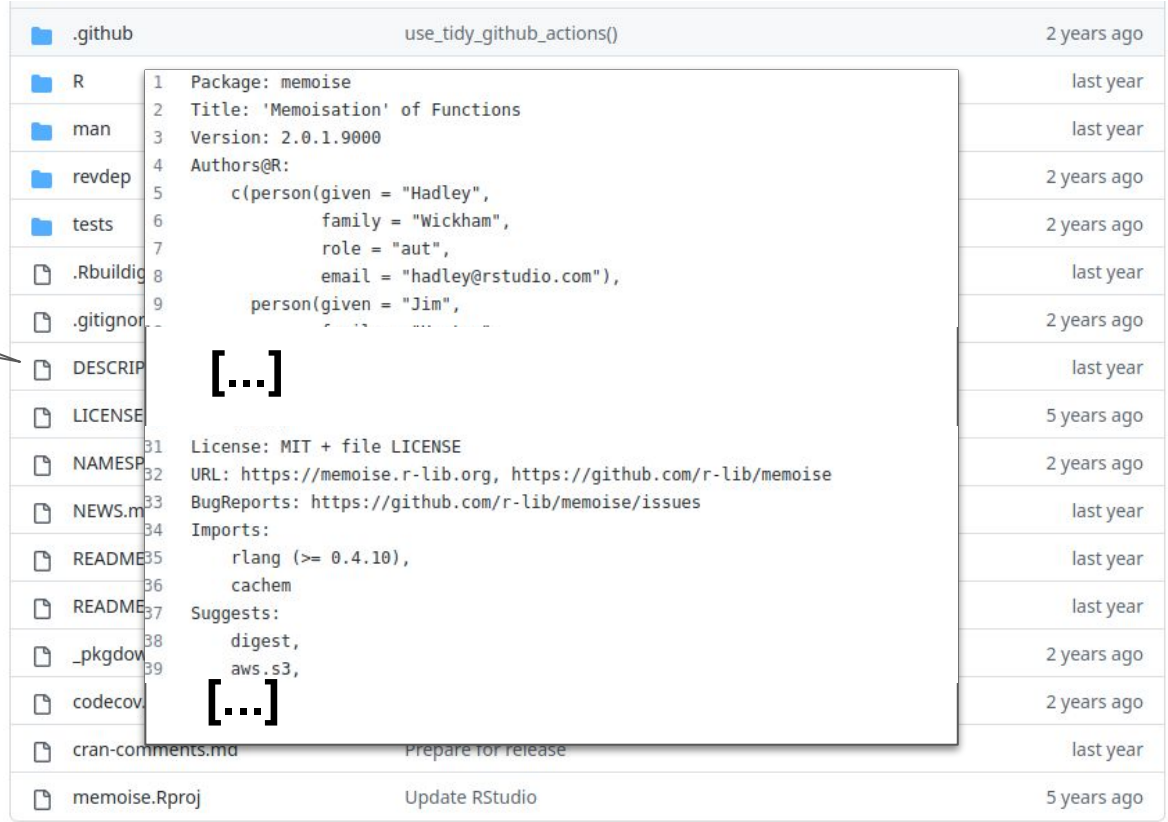


.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Components of an R Package

DESCRIPTION

Meta-info about the package: Name, name of the author, license, package dependencies etc.

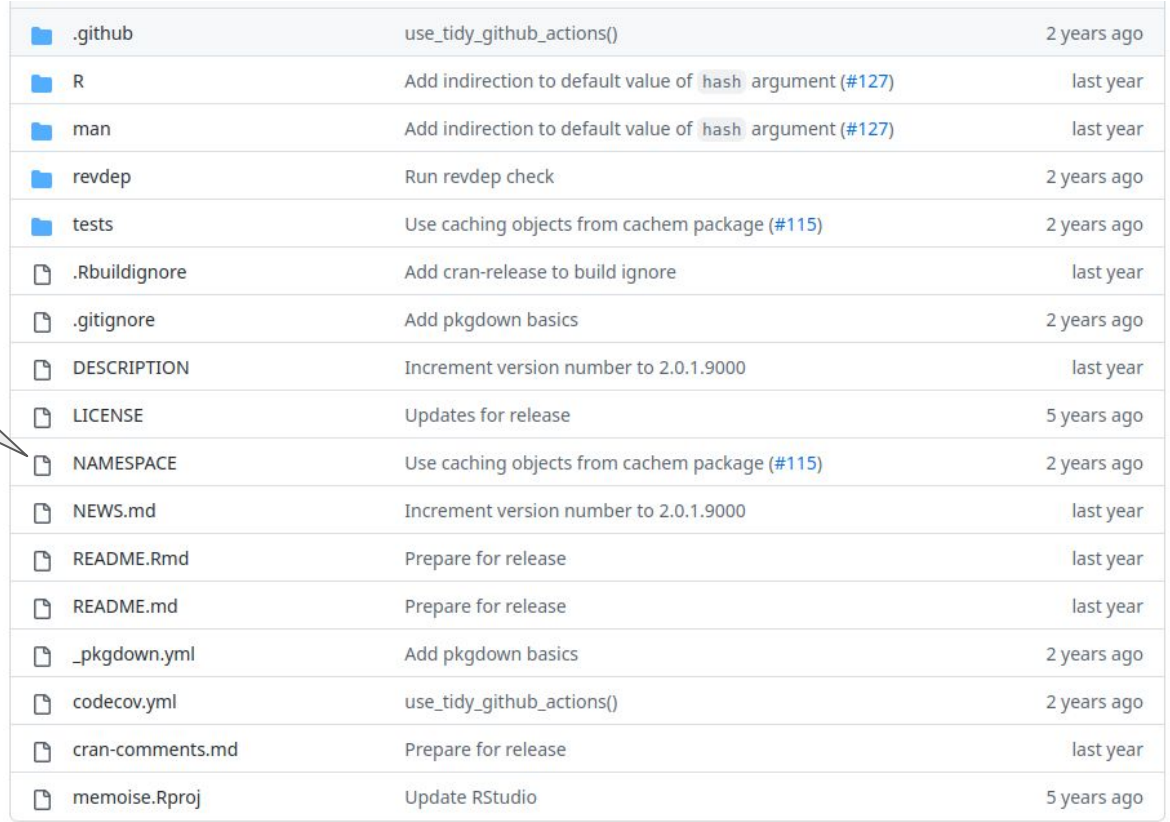


Components of an R Package

NAMESPACE

Imports and exports of the package.

Most people auto-generate this using **roxygen**.



.github	use_tidy_github_actions()	2 years ago
R	Add indirection to default value of `hash` argument (#127)	last year
man	Add indirection to default value of `hash` argument (#127)	last year
revdep	Run revdep check	2 years ago
tests	Use caching objects from cachem package (#115)	2 years ago
.Rbuildignore	Add cran-release to build ignore	last year
.gitignore	Add pkgdown basics	2 years ago
DESCRIPTION	Increment version number to 2.0.1.9000	last year
LICENSE	Updates for release	5 years ago
NAMESPACE	Use caching objects from cachem package (#115)	2 years ago
NEWS.md	Increment version number to 2.0.1.9000	last year
README.Rmd	Prepare for release	last year
README.md	Prepare for release	last year
_pkgdown.yml	Add pkgdown basics	2 years ago
codecov.yml	use_tidy_github_actions()	2 years ago
cran-comments.md	Prepare for release	last year
memoise.Rproj	Update RStudio	5 years ago

Code inside an R Package

main ▾ | memoise / R /

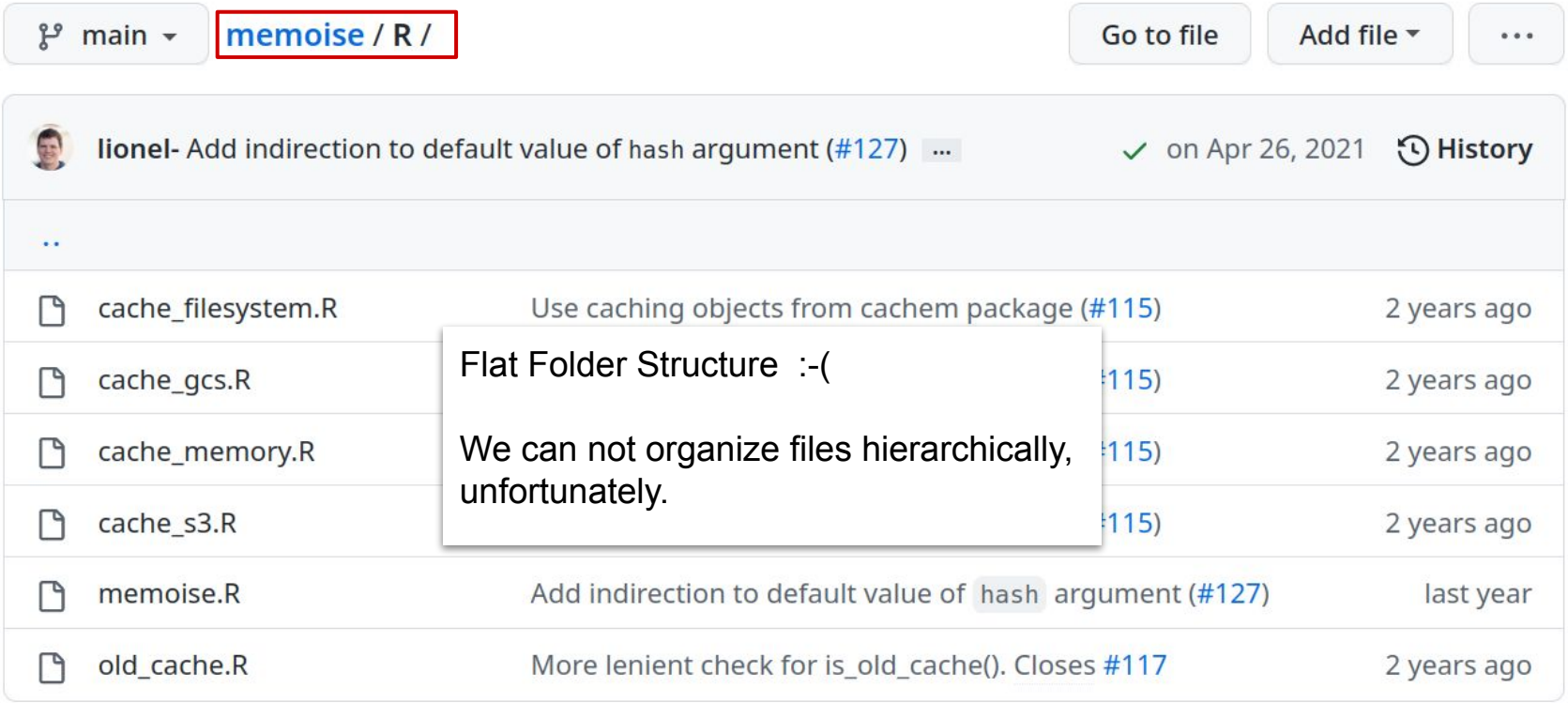
Go to file | Add file ▾ | ···

lionel- Add indirection to default value of `hash` argument (#127) ··· ✓ on Apr 26, 2021 **History**

..

cache_filesystem.R	Use caching objects from cachem package (#115)	2 years ago
cache_gcs.R	Use caching objects from cachem package (#115)	2 years ago
cache_memory.R	Use caching objects from cachem package (#115)	2 years ago
cache_s3.R	Use caching objects from cachem package (#115)	2 years ago
memoise.R	Add indirection to default value of `hash` argument (#127)	last year
old_cache.R	More lenient check for is_old_cache(). Closes #117	2 years ago

Code inside an R Package



Code inside an R Package

```
258  #' Forget past results.
259  #' Resets the cache of a memoised function. Use \code{\link{drop_cache}} to
260  #' reset the cache only for particular arguments.
261  #'
262  #' @param f memoised function
263  #' @export
264  #' @seealso \code{\link{memoise}}, \code{\link{is.memoised}}, \code{\link{drop_cache}}
265  #' @examples
266  #' memX <- memoise(function() { Sys.sleep(1); runif(1) })
267  #' # The forget() function
268  #' system.time(print(memX()))
269  #' system.time(print(memX()))
270  #' forget(memX)
271  #' system.time(print(memX()))
272  forget <- function(f) {
273    if (!is.memoised(f)) {
274      return(FALSE)
275    }
276
277    env <- environment(f)
278    if (!exists("_cache", env, inherits = FALSE)) return(FALSE) # nocovr
279
280    cache <- get("_cache", env)
281    cache$reset()
282
283    TRUE
284  }
```

```
> forget
function (f)
{
    if (!is.memoised(f)) {
        return(FALSE)
    }
    env <- environment(f)
    if (!exists("_cache", env, inherits = FALSE))
        return(FALSE)
    cache <- get("_cache", env)
    cache$reset()
    TRUE
}
<bytecode: 0x5601c4edef88>
<environment: namespace:memoise>
```

Code inside an R Package

```
258  #' Forget past results.
259  #' Resets the cache of a memoised function. Use \code{\link{drop_cache}} to
260  #' reset the cache only for particular arguments.
261  #'
262  #' @param f memoised function
263  #' @export
264  #' @seealso \code{\link{memoise}}, \code{\link{is.memoised}}, \code{\link{drop_cache}}
265  #' @examples
266  #' memX <- memoise(function() { Sys.sleep(1); runif(1) })
267  #' # The forget() function
268  #' system.time(print(memX()))
269  #' system.time(print(memX()))
270  #' forget(memX)
271  #' system.time(print(memX()))
272  forget <- function(f) {
273    if (!is.memoised(f)) {
274      return(FALSE)
275    }
276
277    env <- environment(f)
278    if (!exists("_cache", env, inherits = FALSE)) return(FALSE) # nocovr
279
280    cache <- get("_cache", env)
281    cache$reset()
282
283    TRUE
284  }
```

forget {memoise} R Documentation

Forget past results. Resets the cache of a memoised function. Use drop_cache to reset the cache only for particular arguments.

Description

Forget past results. Resets the cache of a memoised function. Use drop_cache to reset the cache only for particular arguments.

Usage

```
forget(f)
```

Arguments

f	memoised function

See Also

memoise, is.memoised, drop_cache

Examples

```
memX <- memoise(function() { Sys.sleep(1); runif(1) })
# The forget() function
system.time(print(memX()))
system.time(print(memX()))
forget(memX)
system.time(print(memX()))
```

Code inside an R Package

Roxygen-Comments!

```
258  #' Forget past results.
259  #' Resets the cache of a memoised function. Use \code{\link{drop_cache}} to
260  #' reset the cache only for particular arguments.
261  #'
262  #' @param f memoised function
263  #' @export
264  #' @seealso \code{\link{memoise}}, \code{\link{is.memoised}}, \code{\link{drop_cache}}
265  #' @examples
266  #' memX <- memoise(function() { Sys.sleep(1); runif(1) })
267  #' # The forget() function
268  #' system.time(print(memX()))
269  #' system.time(print(memX()))
270  #' forget(memX)
271  #' system.time(print(memX()))
272  forget <- function(f) {
273    if (!is.memoised(f)) {
274      return(FALSE)
275    }
276
277    env <- environment(f)
278    if (!exists("_cache", env, inherits = FALSE)) return(FALSE) # nocovr
279
280    cache <- get("_cache", env)
281    cache$reset()
282
283    TRUE
284  }
```

forget {memoise} R Documentation

Forget past results. Resets the cache of a memoised function. Use drop_cache to reset the cache only for particular arguments.

Description

Forget past results. Resets the cache of a memoised function. Use drop_cache to reset the cache only for particular arguments.

Usage

```
forget(f)
```

Arguments

f	memoised function

See Also

memoise, is.memoised, drop_cache

Examples

```
memX <- memoise(function() { Sys.sleep(1); runif(1) })
# The forget() function
system.time(print(memX()))
system.time(print(memX()))
forget(memX)
system.time(print(memX()))
```

Code inside an R Package

```
258  #' Forget past results.
259  #' Resets the cache of a memoised function. Use \code{\link{drop_cache}} to
260  #' reset the cache only for particular arguments.
261  #'
262  #' @param f memoised function
263  #' @export
264  #' @seealso \code{\link{memoise}}, \code{\link{is.memoised}}, \code{\link{drop_cache}}
265  #' @examples
266  #' memX <- memoise(function() { Sy   leep(1); runif(1) })
267  #' # The forget() function
268  #' system.time(print(memX()))
269  #' system.time(print(memX()))
270  #' forget(memX)
271  #' system.time(print(memX()))
272  forget <- function(f) {
273    if (!is.memoised(f)) {
274      return(FALSE)
275    }
276
277    env <- environment(f)
278    if (!exists("_cache", env, inherits = FALSE)) return(FALSE) # nocovr
279
280    cache <- get("_cache", env)
281    cache$reset()
282
283    TRUE
284  }
```

... Tells roxygen to "export" this function.

main ▾ **memoise / NAMESPACE**

wch Use caching objects from cachem package (#115) ✓

6 contributors

17 lines (16 sloc) | 350 Bytes

```
1  # Generated by roxygen2: do not edit by hand
2
3  S3method(print,memoised)
4  export(cache_filesystem)
5  export(cache_gcs)
6  export(cache_memory)
7  export(cache_s3)
8  export(drop_cache)
9  export(forget)
10 export(has_cache)
11 export(is.memoised)
12 export(is.memoized)
13 export(memoise)
14 export(memoize)
15 export(timeout)
16 importFrom(cachem,key_missing)
17 importFrom(stats,setNames)
```

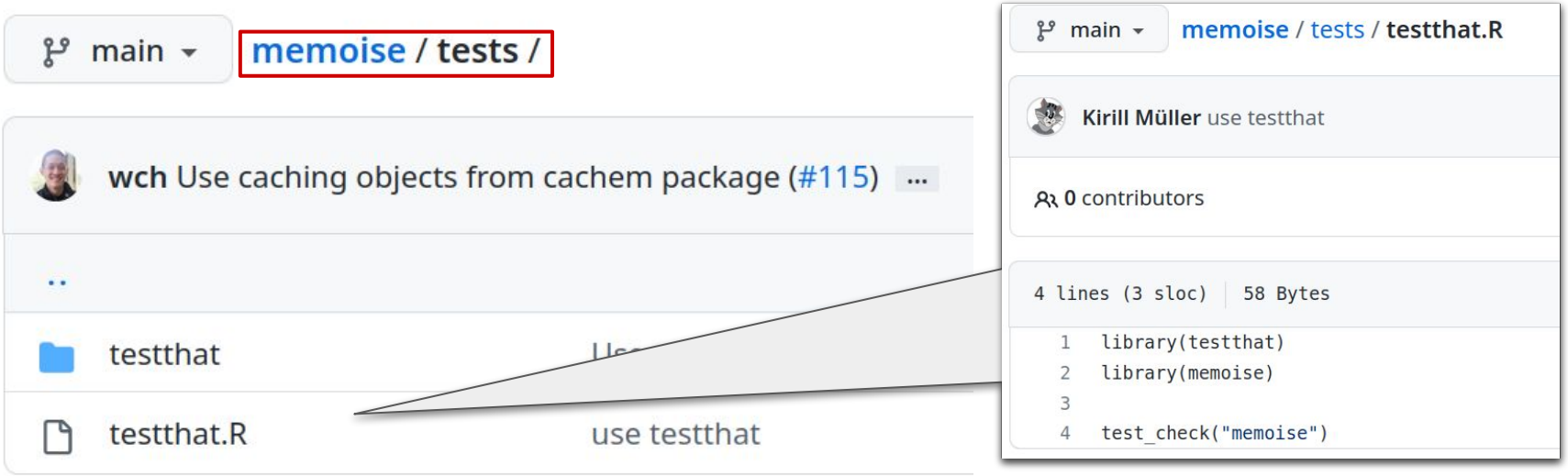
Code inside an R Package

No Roxygen doc, no '@export' --> not exported.

```
26  # Returns TRUE if it's an old-style cache.
27  is_old_cache <- function(x) {
28    is.function(x$digest) &&
29      is.function(x$set) &&
30      is.function(x$get) &&
31      is.function(x$has_key)
32  }
```

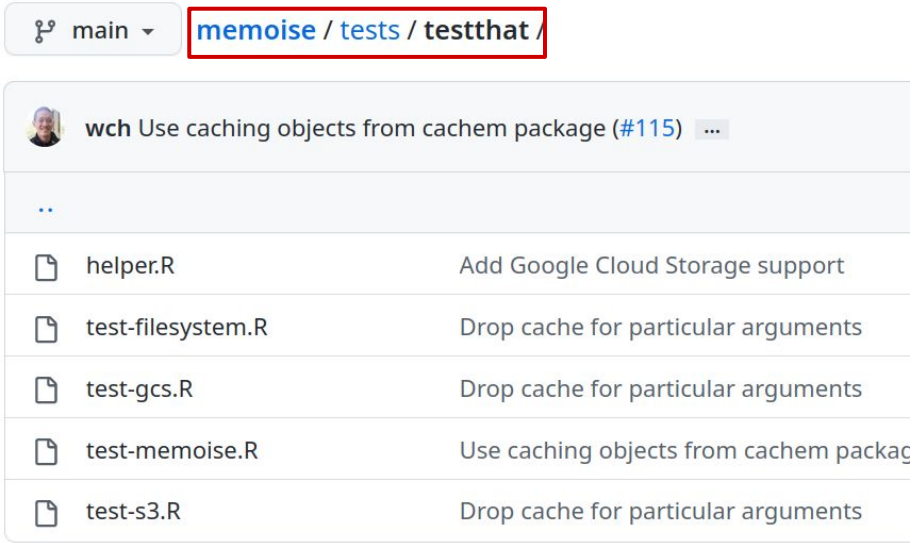
```
> memoise:::is_old_cache
function (x)
{
    is.function(x$digest) && is.function(x$set) && is.function(x$get) &&
        is.function(x$has_key)
}
<bytecode: 0x5601c305f1c0>
<environment: namespace:memoise>
```

Tests of an R Package



- This is "boilerplate code", dictated by the "testthat" package
- There exist other test packages
- Tests can also be done without a test package, just with R-code.

Tests of an R Package



Tests of an R Package

main ▾ **memoise** / tests / testthat / **test-filesystem.R**

richardkunze Drop cache for particular arguments ... Latest cor

2 contributors

41 lines (30 sloc) | 885 Bytes

```
1  context("filesystem")
2
3  test_that("using a filesystem cache works", {
4
5    fs <- cache_filesystem(tempfile())
6    i <- 0
7    fn <- function() { i <<- i + 1; i }
8    fnm <- memoise(fn, cache = fs)
9    on.exit(forget(fnm))
10
```

Test that runs on both CRAN and with the CI system.

Tests of an R Package

main memoise / tests / testthat / test-gcs.R

richardkunze Drop cache for particular arguments ... Latest commit 9

2 contributors

33 lines (26 sloc) 774 Bytes

```
1  context("gcs")
2
3  test_that("using a gcs cache works", {
4    skip_on_cran()
5    skip_on_travis_pr()
6    skip_without_gcs_credentials()
7
8    googleAuthR::gar_set_client(scopes = "https://www.googleapis.com/a
9    googleAuthR::gar_auth_service(Sys.getenv("GCS_AUTH_FILE"))
10
11   aws <- cache_gcs("memoise-tests")
12   i <- 0
13   fn <- function() { i <<- i + 1; i }
```

Test that does not on CRAN but only locally, since it is expensive or undesirable in some way...

How to write a Package

How to Write a Package

https://cran.r-project.org/doc/manuals/R-exts.html

- Folder structure: R, man, tests, (vignettes, inst, src, CI things...)
- Necessary Files: DESCRIPTION, NAMESPACE (auto-generate), README[.md], NEWS[.md] (.Rbuildignore, .gitignore, LICENSE, NEWS.md)
  - Your code, of course!
  - Tests
  - Documentation (auto-generate)

What happens here?

```
> install.packages("checkmate")
Installing package into '/home/user/R/x86_64-redhat-linux-gnu-library/4.1'
(as 'lib' is unspecified)
trying URL 'https://cran.rstudio.com/src/contrib/checkmate_2.1.0.tar.gz'
Content type 'application/x-gzip' length 170034 bytes (166 KB)
==================================================
downloaded 166 KB

* installing *source* package 'checkmate' ...
** package 'checkmate' successfully unpacked and MD5 sums checked
** using staged installation
** libs
gcc -m64 -I"/usr/include/R" -DNDEBUG   -I/usr/local/include   -fpic  -O2  -
 -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -Wp,-D_GLIBCXX_ASSERTIONS
stack-protector-strong -specs=/usr/lib/rpm/redhat/redhat-annobin-cc1  -m64
ack-clash-protection -fcf-protection  -c all_missing.c -o all_missing.o
```

.

.

.

What happens here?

```
.
.
.
nd_nchar.o guess_type.o helper.o init.o integerish.o is_sorted.o qassert.o wh
installing to /home/user/R/x86_64-redhat-linux-gnu-library/4.1/00LOCK-checkma
** R
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
  converting help for package 'checkmate'
    finding HTML links ... done
    AssertCollection                        html
    anyInfinite                             html
    register_test_backend                   html
    vname                                   html
    wf                                      html
** building package indices
** installing vignettes
** testing if installed package can be loaded from temporary location
** checking absolute paths in shared objects and dynamic libraries
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (checkmate)

The downloaded source packages are in
        '/tmp/RtmpSKonrn/downloaded_packages'
```