

Package: cache (via r-universe)

October 24, 2024

Title Cache and Retrieve Computation Results

Version 0.0.3

Description Easily cache and retrieve computation results. The package works seamlessly across interactive R sessions, R scripts and Rmarkdown documents.

URL <https://github.com/OlivierBinette/cache>

BugReports <https://github.com/OlivierBinette/cache/issues>

License MIT + file LICENSE

Imports digest, assert, cli, here

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository <https://olivierbinette.r-universe.dev>

RemoteUrl <https://github.com/olivierbinette/cache>

RemoteRef HEAD

RemoteSha 24988f069a4189e142deda551716036df14db332

Contents

cache	2
cache_load	3
Index	4

cache

Cache or Retrieve Computation Results

Description

Cache or retrieve an evaluated expression. Results are always made available in the current environment.

Usage

```
cache(..., .cachedir = here(".cache-R"), .rerun = FALSE)
```

Arguments

...	Named expressions to be cached or retrieved.
.cachedir	Directory where cache files are stored. Default is a directory called .cache-R located at the project root.
.rerun	Whether or not to clear the cache and re-run the provided expressions. Defaults to FALSE.

Examples

```
tmp <- tempdir()

# Takes 1 second to execute
cache(a = {
  Sys.sleep(1)
  "Hello World"
}, .cachedir = tmp)

# Executes instantly
cache(a = {
  Sys.sleep(1)
  "Hello World"
}, .cachedir = tmp)

# Result is available in the current environment
print(a)

# Re-run the expression
cache(a = {
  Sys.sleep(1)
  "Hello World"
}, .cachedir = tmp, .rerun = TRUE)
```

cache_load	<i>Load cached objects from cache directory</i>
------------	---

Description

Load cached objects from cache directory

Usage

```
cache_load(objnames = "*", .cachedir = here(".cache-R"))
```

Arguments

objnames	character list of object names to load.
.cachedir	path to cache directory. Defaults to the directory named .cache-R at the project root.

Index

cache, [2](#)
cache_load, [3](#)