Package 'reproducibleRchunks'

January 29, 2025

Title Automated Reproducibility Checks for R Markdown Documents

Version 1.0.2

Description Provide reproducible R chunks in R Markdown document that automatically check computational results for reproducibility. This is achieved by creating json files storing metadata about computational results. A comprehensive tutorial to the package is available as preprint by Brandmaier & Peikert (2024, <doi:10.31234/osf.io/3zjvf>).

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.2.3

Imports jsonlite, knitr, digest, rstudioapi, utils

Suggests spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Language en-US

BugReports https://github.com/brandmaier/reproducibleRchunks/issues

NeedsCompilation no

Author Andreas M. Brandmaier [aut, cre],
Aaron Peikert [ctb]

Maintainer Andreas M. Brandmaier <andreas.brandmaier@medicalschool-berlin.de>

Repository CRAN

Date/Publication 2025-01-29 16:40:02 UTC

Contents

get_num_reproducibility_errors

Get the total number of failed reproduction attempts

Description

Get the total number of failed reproduction attempts

Usage

```
get_num_reproducibility_errors()
```

Value

Returns the number of errors encountered when reproducing a Markdown document

get_reproducibility_summary

Get a summary about all reproduction attempts

Description

This function returns a data frame, in which details about reproduction attempts are collected. The data frame has three columns named "Chunk","Variable", and "Success". Every row in the data frame corresponds to one variable, for which reproducibility was tested. `Chunk` stores the name of the surrounding chunk, `Variable` stores the name of the variable, and `Success` is a boolean variable, which indicates whether the reproduction attempt was successful.

Usage

```
get_reproducibility_summary()
```

Value

Returns a data.frame with three columns.

load_repro_data *Loading reproducibility data*

Description

This function loads reproducibility meta data from a file and stores the meta information about the variable contents in the specified environment. Reproducibility meta data can be loaded from either a json (preferred) or a binary saved R object. The function returns a named list with meta information restored from file. The named elements include "hashing" indicating whether a hashing algorithm was used, "hashing_algorithm" indicating the name of the hashing algorithm, "hashing_package" indicating the name of the R package, from which the hashing algorithm was used, "hashing_package_version" indicating the package version, "digits" the numeric precision used before hashing numeric values, and "code_fingerprint" the actual hashed string of the chunk code.

Usage

```
load_repro_data(filename, envir, filetype = c("json", "rda"))
```

Arguments

filename	Character. File name to load objects from.
envir	Environment to load the objects into. By default, this is the global environment.
filetype	Character. Currently supported is json and rda.

Value

Returns a named list with meta information restored from file. See description for more details.

See Also

save_repro_data()

reproducibleR *Knitr Hook*

Description

This is the main RMarkdown chunk hook for processing the automated reproducibility tests of code chunks. This function is not intended to be called directly. Rather, it is expected that RStudio calls this function when rendering chunks with the label identical to this function name.

Usage

```
reproducibleR(options)
```

Arguments

options	A list of chunk options passed from the knitr engine. Usually this is just the object options passed to the engine function; see knit_engines.

Details

This function first executes the R code from a given chunk. If a variable is declared within the scope of the chunk, meta information about the variable's content are generated. If no metadata exists, this metadata is stored in a separate file. If metadata exists, it is compared against the metadata of the reproduction attempt.

Value

A character string generated from the source code and output.

Author(s)

Andreas M. Brandmaier

Examples

```
reproducibleR(knitr::opts_chunk$merge(list(engine="reproducibleR",code="1+1")))
```

---

save_repro_data *Storing reproducibility data*

Description

Storing reproducibility data

Usage

```
save_repro_data(
  x,
  filename,
  filetype = default_filetype(),
  envir = NULL,
  extra = NULL
)
```

Arguments

x	Object to be stored.
filename	Name (possible including full path) of the save file
filetype	Character. Currently supported is json and rda.
envir	Environment to load the objects into. By default, this is the global environment.
extra	List. Extra payload to store in the meta data

Value

No return value

See Also

load_repro_data()

Index