Package 'repmis'

July 3, 2025

Type Package

Title Miscellaneous Tools for Reproducible Research

Description Tools to load 'R' packages
and automatically generate BibTeX files citing them as well as load and
cache plain-text and 'Excel' formatted data stored on 'GitHub', and
from other sources.

Version 0.5.1

Date 2025-07-02

Author Christopher Gandrud [aut, cre]

Maintainer Christopher Gandrud <christopher.gandrud@gmail.com>

License GPL (>= 3)

Depends R (>= 3.0.0)

URL https://CRAN.R-project.org/package=repmis

BugReports https://github.com/christophergandrud/repmis/issues

Imports data.table, digest, httr, plyr, R.cache

Suggests xlsx

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2025-07-02 22:40:02 UTC

Contents

`git_stamp` *Get git stamp (commit and branch) for a repository*

Description

The function returns the latest git commit and branch for the repo specified in `repo` or the current working directory if unspecified. Git is needed for the command to run. The functions makes it possible to include the latest git commit and branch in a run to be able to know exactly which code where used.

Usage

```
git_stamp(repo = getwd())
```

Arguments

`repo`	Git repo directory. If unspecified, then the current working directory is used.

Value

character vector with latest commit and branch

`InstallOldPackages` *Install old versions of R packages.*

Description

`InstallOldPackages` installs specific R package versions.

Usage

```
InstallOldPackages(
  pkgs,
  versions,
  oldRepos = "http://cran.r-project.org",
  lib = NULL
)
```

Arguments

pkgs	character vector of package names to install.
versions	character vector of package version numbers. to install. The order must match the order of package names in pkgs.
oldRepos	character name of repository to download the packages old package versions from. Default is oldRepos = "http://cran.r-project.org".
lib	character vector giving the library directories where to install the packages. Recycled as needed. If NULL, defaults to the first element of .libPaths().

Details

Installs specific R package versions.

See Also

install.packages and download.file

Examples

```
## Not run:
# Install old versions of the e1071 and gtools packages.
Names <- c("e1071", "gtools")
Vers <- c("1.6", "2.6.1")
InstallOldPackages(pkgs = Names, versions = Vers)

## End(Not run)
```

LoadandCite *Install, load, and cite R packages*

Description

LoadandCite can install and load R packages as well as automatically generate a BibTeX file citing the packages.

Usage

```
LoadandCite(
  pkgs = NULL,
  versions = NULL,
  Rversion = NULL,
  bibtex = TRUE,
  style = "plain",
  tweak = TRUE,
  install = FALSE,
  file = NULL,
  repos = NULL,
  lib = NULL
)
```

Arguments

pkgs	a character vector of R package names. If pkgs = NULL then LoadandCite only cites the non-base packages in the current session. It does not load or install any packages.
versions	character vector of package version numbers to install. Only works if install = TRUE. The order must match the order of package names in pkgs.
Rversion	a character string specifying a particular R version. If the version of R currently running differs from Rversion LoadandCite a warning will be given. This argument is for replication purposes.
bibtex	logical. If TRUE than a BibTeX formatted citation file is created. If FALSE than the citations are returned as plain text.
style	character string indicating stylistic elements to add to the citations. Currently supports 'plain', i.e. no special formatting and 'JSS' to match the BibTeX style for the *Journal of Statistical Software* (see https://www.jstatsoft.org/style).
tweak	logical. Whether to fix some known problems in the citations, especially non-standard format of authors.
install	a logical option for whether or not to install the packages. The default is install = FALSE.
file	the name of the BibTeX file you want to create. If file = NULL then the packages are loaded, but no BibTeX file is created.
repos	character vector specifying which repository to download packages from. Only relevant if install = TRUE and versions are not specified. If repos = NULL, automatically reads user defined repository (via options), but defaults to repos = "http://cran.us.r-project.org" if repos is not set.
lib	character vector giving the library directories where to install the packages. Recycled as needed. If NULL, defaults to the first element of .libPaths(). Only relevant if install = TRUE.

Details

The command can install R packages, load them, and create a BibTeX file that can be used to cite the packages in a LaTeX or similar document. It can be useful to place this command in a knitr code chunk at the beginning of a reproducible research document. Note: the command will overwrite existing files with the same name as file, so it is generally a good idea to create a new BibTeX file with LoadandCite.

Source

Gandrud, Christopher (2013). Automating R Package Citations in Reproducible Research Documents. SSRN. This function is partially based on: https://gist.github.com/3710171. It also builds on code from knitr's write_bib. See: Y. Xie. knitr: A general-purpose package for dynamic report generation in R, 2013. URL https://CRAN.R-project.org/package=knitr. R package version 1.5. Note that it does not formally depend on knitr so that knitr can be included in LoadandCite so that it is possible to install old versions of that package.

See Also

write_bib, install.packages, and library

Examples

```
## Not run:
# Create vector of package names
## In this example you need to have the packages installed aready.
PackNames <- "repmis"
# Load the packages and create a BibTeX file
LoadandCite(pkgs = PackNames, file = 'PackageCites.bib', style = 'JSS')

## End(Not run)
## Not run:
# Install, load, and cite specific package versions
# dontrun due to CRAN restrictions
Names <- c("e1071", "gtools")
Vers <- c("1.6", "2.6.1")
LoadandCite(pkgs = Names, versions = Vers, install = TRUE,
             file = "PackageCites.bib")

## End(Not run)
```

---

scan_https *Read a character text file from a secure (https) site into R as a single object.*

Description

Read a character text file from a secure (https) site into R as a single object.

Usage

```
scan_https(url, sha1 = NULL)
```

Arguments

url	The files's URL.
sha1	Character string of the file's SHA-1 hash, generated by source_data. Note if you are using data stored using Git, this is not the file's commit SHA-1 hash.

Value

a charcter object of length 1

Source

Originally based on source_url from the Hadley Wickham's devtools package.

set_valid_wd *Sets valid working directory from vector of possible directories*

Description

Sets valid working directory from vector of possible directories

Usage

```
set_valid_wd(possible)
```

Arguments

possible	character vector of possible working directores

Details

Sets the working directory to the first valid directory from a list of possible directories.

Examples

```
## Not run:
set_valid_wd(c('examples/directory1', 'anotherExample/directory2'))

## End(Not run)
```

source_data *Load plain-text data and RData from a URL (either http or https)*

Description

source_data loads plain-text or RDATA formatted data stored at a URL (both http and https) into R.

Usage

```
source_data(
  url,
  rdata,
  sha1 = NULL,
  cache = FALSE,
  clearCache = FALSE,
  sep = "auto",
  header = "auto",
  stringsAsFactors = FALSE,
```

```
  envir = parent.frame(),
  ...
)
```

Arguments

url	The data's URL. To distinguish between plain-text and RDATA the url must end in a distinguishing file extension.
rdata	logical. Whether or not the data set is an .RDATA file. If not specified than source_url will attempt to determine whether or not the file is an .RDATA file from the URL's extension.
sha1	Character string of the file's SHA-1 hash, generated by source_data. Note if you are using data stored using Git, this is not the file's commit SHA-1 hash.
cache	logical. Whether or not to cache the data so that it is not downloaded every time the function is called.
clearCache	logical. Whether or not to clear the downloaded data from the cache.
sep	The separator method for the plain-text data. For example, to load comma-separated values data (CSV) use sep = ",". To load tab-separated values data (TSV) use sep = "\t". Only relevant for plain-text data.
header	Logical, whether or not the first line of the file is the header (i.e. variable names).
stringsAsFactors	logical. Convert all character columns to factors?
envir	the environment where the data should be loaded.
...	additional arguments passed to fread or load as relevant.

Details

Loads plain-text data (e.g. CSV, TSV) or RDATA from a URL. Works with both HTTP and HTTPS sites. Note: the URL you give for the url argument must be for the RAW version of the file. The function should work to download plain-text data from any secure URL (https), though I have not verified this.

From the source_url documentation: "If a SHA-1 hash is specified with the sha1 argument, then this function will check the SHA-1 hash of the downloaded file to make sure it matches the expected value, and throw an error if it does not match. If the SHA-1 hash is not specified, it will print a message displaying the hash of the downloaded file. The purpose of this is to improve security when running remotely-hosted code; if you have a hash of the file, you can be sure that it has not changed."

Value

a data frame

Source

Originally based on source_url from the Hadley Wickham's devtools package.

See Also

GET, fread, and load

Examples

```
## Not run:
# Download electoral disproportionality data stored on GitHub
# Note: Using shortened URL created by bitly
DisData <- source_data("http://bit.ly/156oQ7a")

# Check to see if SHA-1 hash matches downloaded file
DisDataHash <- source_data("http://bit.ly/Ss6zDO",
   sha1 = "dc8110d6dff32f682bd2f2fdbacb89e37b94f95d")

## End(Not run)
```

---

source_DropboxData *No longer supported*

Description

No longer supported

Usage

```
source_DropboxData()
```

---

source_XlsxData *Download an Excel data set*

Description

source_XlsxData loads Excel data stored at a URL (both http and https) into R.

Usage

```
source_XlsxData(
  url,
  sheet,
  sha1 = NULL,
  cache = FALSE,
  clearCache = FALSE,
  ...
)
```

Arguments

`url`	character string of the Excel files's URL.
`sheet`	character string of number of representing the sheet in the workbook to return. Only one sheet at a time can currently be returned
`sha1`	Character string of the file's SHA-1 hash, generated by `source_data`. Note if you are using data stored using Git, this is not the file's commit SHA-1 hash.
`cache`	logical. Whether or not to cache the data so that it is not downloaded every time the function is called.
`clearCache`	logical. Whether or not to clear the downloaded data from the cache.
`...`	arguments to pass to `read.xlsx` from the xlsx package.

Value

a data frame

See Also

`read.xlsx`, `GET`, `source_data`

Index