Package: tidytuesdayR (via r-universe)

September 17, 2024

Type Package

Title Access the Weekly 'TidyTuesday' Project Dataset

Version 1.1.2.9000

Description 'TidyTuesday' is a project by the 'Data Science Learning Community' in which they post a weekly dataset in a public data repository (<https://github.com/rfordatascience/tidytuesday>) for people to analyze and visualize. This package provides the tools to easily download this data and the description of the source.

License MIT + file LICENSE

URL https://dslc-io.github.io/tidytuesdayR/, https://github.com/dslc-io/tidytuesdayR

BugReports https://github.com/dslc-io/tidytuesdayR/issues

Depends R (>= 3.5.0)

Imports cli, gh, glue, jsonlite, lubridate (>= 1.7.0), magrittr, purrr (>= 1.0.0), readr (>= 1.0.0), rlang, rvest (>= 0.3.2), tidyr, tools (>= 3.1.0), usethis, xml2 (>= 1.2.0)

Suggests covr, pkgdown, readxl (>= 1.0.0), rstudioapi (>= 0.2), stringr, testthat (>= 3.0.0), tibble, withr

Config/testthat/edition 3

Encoding UTF-8

Language en-US

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository https://dslc-io.r-universe.dev

RemoteUrl https://github.com/dslc-io/tidytuesdayr

RemoteRef HEAD

RemoteSha a6ab3546466a99dfff5a11dbf2c5185942e5f58a

Contents

`last_tuesday` *Find the most recent tuesday*

Description

Identify the most recent 'TidyTuesday' date relative to a specified date.

Usage

```
last_tuesday(date = today(tzone = "America/New_York"))
```

Arguments

`date`	A date as a date object or character string in `YYYY-MM-DD` format. Defaults to today's date.

Value

The TidyTuesday date in the same week as the specified date, using Monday as the start of the week.

Examples

```
last_tuesday() # get last Tuesday relative to today's date
last_tuesday("2020-01-01") # get last Tuesday relative to a specified date
```

print.tt_data *print methods of the tt objects*

Description

In tidytuesdayR there are nice print methods for the objects that were used to download and store the data from the TidyTuesday repo. They will always print the available datasets/files. If there is a readme available, it will try to display the TidyTuesday readme.

Usage

```
## S3 method for class 'tt_data'
print(x, ...)

## S3 method for class 'tt'
print(x, ...)
```

Arguments

x	a tt_data or tt object
...	further arguments passed to or from other methods.

Value

used to show readme and list names of available datasets

x, invisibly.

Examples

```
tt <- tt_load_gh("2019-01-15")
print(tt)

tt_data <- tt_download(tt, files = "All")
print(tt_data)
```

readme *Readme HTML maker and Viewer*

Description

Readme HTML maker and Viewer

Usage

```
readme(tt)
```

Arguments

tt	tt_data object for printing

Value

Null, invisibly. Used to show readme of the downloaded TidyTuesday dataset in the Viewer.

Examples

```
if (rate_limit_check(quiet = TRUE) > 30) {
  tt_output <- tt_load_gh("2019-01-15")
  readme(tt_output)
}
```

---

tt_available *Listing all available TidyTuesdays*

Description

The TidyTuesday project is a constantly growing repository of data sets. Knowing what type of data is available for each week requires going to the source. However, one of the hallmarks of 'tidytuesdayR' is that you never have to leave your R console. These functions were created to help maintain this philosophy.

Usage

```
tt_available(auth = gh::gh_token())

tt_datasets(year, auth = gh::gh_token())
```

Arguments

auth	A GitHub token. See gh::gh_token() for more details.
year	What year of TidyTuesday to use

Details

To find out the available datasets for a specific year, the user can use the function `tt_datasets()`. This function will either populate the Viewer or print to console all the available data sets and the week/date they are associated with.

To get the whole list of all the data sets ever released by TidyTuesday, the function `tt_available()` was created. This function will either populate the Viewer or print to console all the available data sets ever made for TidyTuesday.

Value

`tt_available()` returns a `tt_dataset_table_list`, which is a list of `tt_dataset_table`. This class has special printing methods to show the available data sets.

`tt_datasets()` returns a `tt_dataset_table` object. This class has special printing methods to show the available datasets for the year.

Examples

```
# check to make sure there are requests still available
if (rate_limit_check(quiet = TRUE) > 30) {
  ## show data available from 2018
  tt_datasets(2018)

  ## show all data available ever
  tt_available()
}
```

`tt_download` *Download TidyTuesday data*

Description

Download all or specific files identified in a TidyTuesday dataset.

Usage

```
tt_download(tt, files = "All", ..., auth = gh::gh_token())
```

Arguments

`tt`	A `tt` object, output from `tt_load_gh()`.
`files`	Which file names to download. Default "All" downloads all files for the specified week.
`...`	Additional parameters to pass to the parsing functions. Note: These arguments will be passed for all filetypes.
`auth`	A GitHub token. See `gh::gh_token()` for more details.

Value

A list of tibbles from the downloaded files.

Examples

```
# Get the list of files for a week.
tt_output <- tt_load_gh("2019-01-15")

# Download a specific file.
agencies <- tt_download(tt_output, files = "agencies.csv")
```

`tt_download_file` *Download a TidyTuesday dataset file*

Description

Download an actual data file from the TidyTuesday github repository.

Usage

```
tt_download_file(tt, x, ..., auth = gh::gh_token())
```

Arguments

`tt`	A `tt` object, output from `tt_load_gh()`.
`x`	Index or name of file to download.
`...`	Additional parameters to pass to the parsing functions. Note: These arguments will be passed for all filetypes.
`auth`	A GitHub token. See `gh::gh_token()` for more details.

Value

tibble containing the contents of the file downloaded from git

Examples

```
tt_gh <- tt_load_gh("2019-01-15")

agencies <- tt_download_file(tt_gh, 1)
launches <- tt_download_file(tt_gh, "launches.csv")
```

tt_load *Load TidyTuesday data from Github*

Description

Load TidyTuesday data from Github

Usage

```
tt_load(x, week = NULL, files = "All", ..., auth = gh::gh_token())
```

Arguments

x	The date of data to pull (in "YYYY-MM-dd" format), or the four-digit year as a number.
week	Which week number to use within a given year. Only used when x is a valid year.
files	Which file names to download. Default "All" downloads all files for the specified week.
...	Additional parameters to pass to the parsing functions. Note: These arguments will be passed for all filetypes.
auth	A GitHub token. See gh::gh_token() for more details.

Value

tt_data object, which contains data that can be accessed via $, and the readme for the week's TidyTuesday, which can be viewed by printing the object or calling readme().

Examples

```
tt_output <- tt_load("2019-01-15")
tt_output
agencies <- tt_output$agencies
```

tt_load_gh *Load TidyTuesday data from Github*

Description

Pulls the readme and URLs of the data from the TidyTuesday github folder based on the date provided

Usage

```
tt_load_gh(x, week = NULL, auth = gh::gh_token())
```

Arguments

x	The date of data to pull (in "YYYY-MM-dd" format), or the four-digit year as a number.
week	Which week number to use within a given year. Only used when x is a valid year.
auth	A GitHub token. See gh::gh_token() for more details.

Value

A tt object. This contains the files available for the week, readme html, and the date of the TidyTuesday.

Examples

```
# check to make sure there are requests still available
if (rate_limit_check(quiet = TRUE) > 30) {
  tt_gh <- tt_load_gh("2019-01-15")
  ## readme attempts to open the readme for the weekly dataset
  readme(tt_gh)

  agencies <- tt_download(
    tt_gh,
    files = "agencies.csv"
  )
}
```

---

tt_print *Printing Utilities for Listing Available Datasets*

Description

printing utilities for showing the available datasets for a specific year or all time

Usage

```
## S3 method for class 'tt_dataset_table'
print(x, ..., is_interactive = interactive())

## S3 method for class 'tt_dataset_table_list'
print(x, ..., is_interactive = interactive())
```

Arguments

x	an object used to select a method.
...	further arguments passed to or from other methods.
is_interactive	Whether the function is being used interactively.

Value

x, invisibly

Examples

```
# check to make sure there are requests still available
if (rate_limit_check(quiet = TRUE) > 30) {
  available_datasets_2018 <- tt_datasets(2018)
  print(available_datasets_2018)

  all_available_datasets <- tt_available()
  print(all_available_datasets)
}
```

---

use_tidytemplate *Call and open the tidytemplate*

Description

Use the tidytemplate Rmd for starting your analysis with a leg up for processing

Usage

```
use_tidytemplate(
  name = NULL,
  open = rlang::is_interactive(),
  refdate = today(),
  ignore = FALSE
)
```

Arguments

name	A name for your generated TidyTuesday analysis Rmd, such as "My_TidyTuesday.Rmd".
open	Open the newly created file for editing? Happens in RStudio, if applicable, or via `utils::file.edit()` otherwise.
refdate	Date to use as reference to determine which TidyTuesday to use for the template. Either date object or character string in YYYY-MM-DD format.
ignore	Should the newly created file be added to `.Rbuildignore`?

Value

A logical vector indicating whether the file was created or modified, invisibly.

Examples

```
use_tidytemplate(name = "My_Awesome_TidyTuesday.Rmd")
```

Index