Package 'tidytuesdayR'

June 15, 2020

Type Package

Title Access the Weekly 'TidyTuesday' Project Dataset

Version 1.0.0

Description

'TidyTuesday' is a project by the 'R4DS Online Learning Community' in which they post a weekly dataset onto 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://github.com/thebioengineer/tidytuesdayR

BugReports https://github.com/thebioengineer/tidytuesdayR/issues

Encoding UTF-8

LazyData true

RoxygenNote 7.1.0

Depends R (¿= 3.4.0)

Suggests testthat (¿= 2.1.0), covr, pkgdown, tibble

Imports readxl (¿= 1.0.0), rvest (¿= 0.3.2), tools (¿= 3.1.0), lubridate (¿= 1.7.0), purrr (¿= 0.2.5), readr (¿= 1.0.0), rstudioapi (¿= 0.2), xml2 (¿= 1.2.0), httr, jsonlite, magrittr

NeedsCompilation no

Author Ellis Hughes [aut, cre],
Jon Harmon [ctb],
Thomas Mock [ctb],
R4DS Online Learning Community [dtc]

Maintainer Ellis Hughes <ellishughes@live.com>

R topics documented:

---

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 = github_pat())

tt_datasets(year, auth = github_pat())
```

Arguments

auth	github Personal Access Token. See PAT section for more information
year	numeric entry representing the year of tidytuesday you want the list of datasets for. Leave empty for most recent year.

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.

PAT

A Github PAT is a Personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions at ¡https://happygitwithr.com/github-pat.html¿ to set your PAT.

Examples

```
# check to make sure there are requests still available
if(rate_limit_check(silent = TRUE) > 10){
 ## show data available from 2018
 tt_datasets(2018)

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

Available_Printing   *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	is the console interactive?

Value

used for side effects to show the available datasets for the year or for all time.

Examples

```
# check to make sure there are requests still available
if(rate_limit_check(silent = TRUE) > 10){

 available_datasets_2018 <- tt_datasets(2018)
 print(available_datasets_2018)

 all_available_datasets <- tt_available()
 print(all_available_datasets)

}
```

---

github_pat   *Return the local user's GitHub Personal Access Token*

Description

Extract the GitHub Personal Access Token (PAT) from the system environment for authenticated requests.

Usage

```
github_pat(quiet = TRUE)
```

Arguments

quiet   Should this be loud? default TRUE.

Value

a character vector that is your Personal Access Token, or NULL

PAT

A Github 'PAT' is a Personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions from ¡https://happygitwithr.com/github-pat.html¿ to set the PAT.

Examples

```
## if you have a personal access token saved, this will return that value
github_pat()
```

---

printing	*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

used to show available datasets for the tidytuesday

Examples

```
## Not run:

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

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


## End(Not run)
```

---

rate_limit_check   *Get Rate limit left for GitHub Calls*

Description

The GitHub API limits the number of requests that can be sent within an hour. This function returns the stored rate limits that are remaining.

Usage

```
rate_limit_check(n = 10, quiet = TRUE, silent = FALSE)
```

Arguments

n	number of requests that triggers a warning indicating the user is close to the limit
quiet	should the only an error be thrown when the rate limit is zero?
silent	should no warnings or errors be thrown and only the value returned?

Value

return the number of calls are remaining as a numeric values

Examples

```
rate_limit_check(silent = TRUE)
```

---

readme   *Readme HTML maker and Viewer*

Description

Readme HTML maker and Viewer

Usage

```
readme(tt)
```

Arguments

tt	tt_data object for printing

Value

Does not return anything. Used to show readme of the downloaded tidytuesday dataset in the Viewer.

Examples

```
## Not run:
tt_output <- tt_load_gh("2019-01-15")
readme(tt_output)

## End(Not run)
```

---

tt_download	*download tt data Download all or specific files identified in the tt dataset*

Description

download tt data

Download all or specific files identified in the tt dataset

Usage

```
tt_download(tt, files = c("All"), ..., branch = "master", auth = github_pat())
```

Arguments

tt	string representation of the date of data to pull, in YYYY-MM-dd format, or just numeric entry for year
files	List the file names to download. Default to asking.
...	pass methods to the parsing functions. These will be passed to ALL files, so be careful.
branch	which branch to be downloading data from. Default and always should be "master".
auth	github Personal Access Token. See PAT section for more information

Value

list of tibbles of the files downloaded.

PAT

A Github PAT is a personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions at ¡https://happygitwithr.com/github-pat.html¿ to set the PAT.

Examples

```
## Not run:
tt_output <- tt_load_gh("2019-01-15")
agencies <- tt_download(tt_output, files = "agencies.csv")

## End(Not run)
```

---

tt_download_file   *Reads in TidyTuesday datasets from Github repo*

Description

Reads in the actual data from the TidyTuesday github

Usage

```
tt_download_file(tt, x, ..., auth = github_pat())
```

Arguments

tt	tt_gh object from tt_load_gh function
x	index or name of data object to read in. string or int
...	pass methods to the parsing functions. These will be passed to ALL files, so be careful.
auth	github Personal Access Token. See PAT section for more information

Value

tibble containing the contents of the file downloaded from git

PAT

A Github PAT is a personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions at ¡https://happygitwithr.com/github-pat.html¿ to set the PAT.

Examples

```
## Not run:
tt_gh <- tt_load_gh("2019-01-15")

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

## End(Not run)
```

---

tt_load   *Load TidyTuesday data from Github*

Description

Load TidyTuesday data from Github

Usage

```
tt_load(x, week, download_files = "All", ..., auth = github_pat())
```

Arguments

x	string representation of the date of data to pull, in YYYY-MM-dd format, or just numeric entry for year
week	left empty unless x is a numeric year entry, in which case the week of interest should be entered
download_files	which files to download from repo. defaults and assumes "All" for the week.
...	pass methods to the parsing functions. These will be passed to ALL files, so be careful.
auth	github Personal Access Token. See PAT section for more information

Value

tt_data object, which contains data that can be accessed via '$', and the readme for the weeks tidytuesday through printing the object or calling 'readme()'

PAT

A Github PAT is a personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions from ¡https://happygitwithr.com/github-pat.html¿ to set the PAT.

Examples

```
# check to make sure there are requests still available
if(rate_limit_check(silent = TRUE) > 10){

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, auth = github_pat())
```

Arguments

x	string representation of the date of data to pull, in YYYY-MM-dd format, or just numeric entry for year
week	left empty unless x is a numeric year entry, in which case the week of interest should be entered
auth	github Personal Access Token. See PAT section for more information

Value

a 'tt' object. This contains the files available for the week, readme html, and the date of the tidytuesday.

PAT

A Github PAT is a personal Access Token. This allows for signed queries to the github api, and increases the limit on the number of requests allowed from 60 to 5000. Follow instructions from ¡https://happygitwithr.com/github-pat.html¿ to set the PAT.

Examples

```
# check to make sure there are requests still available
if(rate_limit_check(silent = TRUE) > 10){
 tt_gh <- tt_load_gh("2019-01-15")
 readme(tt_gh)
}
```