councilverse

Overview

The `councilverse` is a set of packages that work in tandem to assist the NYCC’s data team in doing their work. This package is designed to make it easy to install and load multiple ‘councilverse’ packages in a single step.

Installation

You can install the released version of `councilverse` from GitHub

```
remotes::install_github("newyorkcitycouncil/councilverse")
```

Load Package

Note that the order of loading the libraries is important. Make sure to load `councilverse` last.

```
library(tidyverse)
# load last
library(councilverse)
```

Vignette

For a demo of the 2 largest packages (`councildown` and `councilcount`) available in `councilverse` , see `vignettes/councilverse.Rmd`

Quick Start

First load the `councilverse` package as above.

Councildown

The `councildown` package implements style guide compliant defaults for R Markdown documents, `ggplot2` plots, and `leaflet` maps. For more information, please visit the package directly: https://github.com/NewYorkCityCouncil/councildown/ . The `councilverse` vignette goes through an example using most of `councildown` ’s available functions.

Councilcount

The `councilcount` package allows easy access to ACS population data across various geographic boundaries. For the boundaries that are not native to the ACS, such as council districts, an estimate is provided. For more information, please visit the package directly: https://github.com/NewYorkCityCouncil/councilcount/ . The `councilverse` vignette goes through an example using most of `councilcount` ’s available functions.

Other Functions

The following functions are standalone functions in `councilverse` used in more niche situations.

Graphs and Tables

`encodeGraphic()` helps convert images to raw HTML. This is useful for adding extra graphics to any HTML element in visualizations. `gt_table()` similarly converts `gt` outputs to raw HTML.

An example of both these functions being used can be found in the NYCHA Scrape repo: https://newyorkcitycouncil.github.io/NYCHA_Scrape/visualization/heat_outage_scatter.html

Misc

`file_name_generator()` creates file names using accepted guidelines detailed in the function’s documentation.

```
# All fields used
file_name_generator(order = "01", description = "acs_poverty", time_granularity = "da
#> [1] "01_acs-poverty_daily_by-cd-race_2018_example.R"
```

`unzip_sf()` loads in shapefiles that come originally as zip files.

```
url <- "https://www1.nyc.gov/assets/planning/download/zip/data-maps/open-data/nyct202
# unzip the zip file
zip <- unzip_sf(url)
# assign to variable with read_csv or other read functions
sf <- sf::read_sf(zip)
```