Package ‘refuge’

October 14, 2022

Version 0.3.3
Title Locate Trans and Intersex-Friendly Toilets
Description Access the ‘Refuge’ API, a web-application for locating trans and intersex-friendly restrooms, including unisex and accessible restrooms. Includes data on the location of restrooms, along with directions, comments, user ratings and amenities. Coverage is global, but data is most comprehensive in the United States. See <https://www.refugerestrooms.org/api/docs/> for full API documentation.
License MIT + file LICENSE
Encoding UTF-8
LazyData true
ByteCompile true
Imports jsonlite, tibble, dplyr, httr
RoxygenNote 7.1.1
URL https://docs.evanodell.com/refuge
BugReports https://github.com/evanodell/refuge/issues
Suggests covr, testthat, knitr, rmarkdown
VignetteBuilder knitr
NeedsCompilation no
Author Evan Odell [aut, cre] (<https://orcid.org/0000-0003-1845-808X>)
Maintainer Evan Odell <evanodell191@gmail.com>
Repository CRAN
Date/Publication 2021-01-11 19:00:02 UTC

R topics documented:

rfg_all_restrooms .................................................. 2
rfg_date ............................................................... 3
rfg_location ......................................................... 4
rfg_search .......................................................... 5
Description

All listed refuge restrooms, ordered by date added or last updated, with the most recent additions and updates at the top.

Usage

```r
rfg_all_restrooms(
  accessible = FALSE,
  unisex = FALSE,
  verbose = TRUE,
  tidy = FALSE
)
```

Arguments

- **accessible**
  - If TRUE, only returns restrooms that meet the Americans with Disability Act standards for accessibility. Defaults to FALSE.
- **unisex**
  - If TRUE, only returns unisex restrooms. Defaults to FALSE.
- **verbose**
  - If TRUE, prints query progress. Defaults to TRUE.
- **tidy**
  - If TRUE, makes USA state names more consistent. The internal function focuses on correcting known errors and mistakes, if incorrect state data is supplied that cannot be corrected. Defaults to FALSE.

Details

Note that the API does not standardise or validate the names of states, cities, countries, etc, and so this data may need to be cleaned and organised.

Value

A tibble with details on all listed bathrooms.

Examples

```r
## Not run:
a <- rfg_all_restrooms(accessible = TRUE, unisex = TRUE)

## End(Not run)
```
### rfg_date

**Bathrooms by dates**

#### Description

Requests all records on bathrooms created or updated on or after a given date.

#### Usage

```r
rfg_date(
  date = NULL,
  accessible = FALSE,
  unisex = FALSE,
  updated = FALSE,
  verbose = TRUE,
  tidy = FALSE
)
```

#### Arguments

- **date**
  A date in "yyyy-mm-dd" format, or any format that can be coerced to a date with `as.Date()`.

- **accessible**
  If TRUE, only returns restrooms that meet the Americans with Disability Act standards for accessibility. Defaults to FALSE.

- **unisex**
  If TRUE, only returns unisex restrooms. Defaults to FALSE.

- **updated**
  If TRUE, returns all bathrooms updated or added since the given date. Defaults to FALSE, which only returns bathrooms added since the given date.

- **verbose**
  If TRUE, prints query progress. Defaults to TRUE.

- **tidy**
  If TRUE, makes USA state names more consistent. The internal function focuses on correcting known errors and mistakes, if incorrect state data is supplied that cannot be corrected. Defaults to FALSE.

#### Value

A tibble with all bathrooms recorded or updated on or after the given date.

#### Examples

```r
## Not run:
q <- rfg_date("2017-11-04")

## End(Not run)
```
rfg_location

Bathroom locations

Description
Given a pair of coordinates (latitude and longitude), finds any nearby bathrooms, subject to other parameters.

Usage
```r
rfg_location(
  lat,
  lng,
  accessible = FALSE,
  unisex = FALSE,
  verbose = TRUE,
  tidy = FALSE
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lat</td>
<td>Location latitude</td>
</tr>
<tr>
<td>lng</td>
<td>Location longitude</td>
</tr>
<tr>
<td>accessible</td>
<td>If TRUE, only returns restrooms that meet the Americans with Disability Act standards for accessibility. Defaults to FALSE.</td>
</tr>
<tr>
<td>unisex</td>
<td>If TRUE, only returns unisex restrooms. Defaults to FALSE.</td>
</tr>
<tr>
<td>verbose</td>
<td>If TRUE, prints query progress. Defaults to TRUE.</td>
</tr>
<tr>
<td>tidy</td>
<td>If TRUE, makes USA state names more consistent. The internal function focuses on correcting known errors and mistakes, if incorrect state data is supplied that cannot be corrected. Defaults to FALSE.</td>
</tr>
</tbody>
</table>

Value
A tibble with details of any and all nearby refuge bathrooms.

Examples
```r
## Not run:
c <- rfg_location(lat = 39, lng = -75)
## End(Not run)
```
Description

Restroom record search

Usage

rgf_search(
    search,
    accessible = FALSE,
    unisex = FALSE,
    verbose = TRUE,
    tidy = FALSE
)

Arguments

search A string to search for. Note that the API does not accept wildcard searches, searches all fields in all available records, and only returns exact (case-insensitive) whole-word matches.

accessible If TRUE, only returns restrooms that meet the Americans with Disability Act standards for accessibility. Defaults to FALSE.

unisex If TRUE, only returns unisex restrooms. Defaults to FALSE.

verbose If TRUE, prints query progress. Defaults to TRUE.

tidy If TRUE, makes USA state names more consistent. The internal function focuses on correcting known errors and mistakes, if incorrect state data is supplied that cannot be corrected. Defaults to FALSE.

Value

A tibble with all bathrooms matching your search request

Examples

## Not run:
s <- rfg_search(search = "toronto")

## End(Not run)
Index

rfg_all_restrooms, 2
rfg_date, 3
rfg_location, 4
rfg_search, 5