

Package ‘nycOpenData’

April 11, 2026

Type Package

Title Convenient Access to NYC Open Data API Endpoints

Version 0.2.1

Description Provides a unified set of helper functions to access datasets from the NYC Open Data platform <<https://opendata.cityofnewyork.us/>>. Functions return results as tidy tibbles and support optional filtering, sorting, and row limits via the Socrata API. The package includes endpoints for 311 service requests, DOB job applications, juvenile justice metrics, school safety, environmental data, event permitting, and additional citywide datasets.

License MIT + file LICENSE

Encoding UTF-8

RoxygenNote 7.3.3

Imports httr, jsonlite, tibble, janitor, curl, dplyr, rlang

Suggests ggplot2, knitr, rmarkdown, scales, testthat (>= 3.0.0), tidyr, vcr (>= 0.6.0), webmockr

URL <https://martinezc1.github.io/nycOpenData/>,
<https://github.com/martinezc1/nycOpenData>

BugReports <https://github.com/martinezc1/nycOpenData/issues>

VignetteBuilder knitr

Config/testthat/edition 3

Depends R (>= 4.1.0)

NeedsCompilation no

Author Christian Martinez [aut, cre] (GitHub: martinezc1, ORCID: <<https://orcid.org/0009-0005-6026-6454>>),
Crystal Adote [ctb] (GitHub: crystalna20),
Jonah Dratfield [ctb] (GitHub: jdratfield38),
Joyce Escatel-Flores [ctb] (GitHub: JoyceEscatel),
Rob Hutto [ctb] (GitHub: robhutto),
Isley Jean-Pierre [ctb] (GitHub: ijpier),
Shannon Joyce [ctb] (GitHub: shannonjoyce),

Laura Rose-Werner [ctb] (GitHub: laurarosewerner),
 Emma Tupone [ctb] (GitHub: emmatup0205),
 Xinru Wang [ctb] (GitHub: xrwangxr)

Maintainer Christian Martinez <c.martinez0@outlook.com>

Repository CRAN

Date/Publication 2026-04-11 18:40:02 UTC

Contents

| | |
|-----------------------------|---|
| nyc_any_dataset | 2 |
| nyc_list_datasets | 3 |
| nyc_pull_dataset | 3 |

| | |
|--------------|----------|
| Index | 6 |
|--------------|----------|

| | |
|-----------------|---------------------------------------|
| nyc_any_dataset | <i>Load Any NYC Open Data Dataset</i> |
|-----------------|---------------------------------------|

Description

Downloads any NYC Open Data dataset given its Socrata JSON endpoint.

Usage

```
nyc_any_dataset(  
  json_link,  
  limit = 10000,  
  timeout_sec = 30,  
  clean_names = TRUE,  
  coerce_types = TRUE  
)
```

Arguments

| | |
|--------------|--|
| json_link | A Socrata dataset JSON endpoint URL (e.g., "https://data.cityofnewyork.us/resource/abcd-1234.json"). |
| limit | Number of rows to retrieve (default = 10,000). |
| timeout_sec | Request timeout in seconds (default = 30). |
| clean_names | Logical; if TRUE, convert column names to snake_case (default = TRUE). |
| coerce_types | Logical; if TRUE, attempt light type coercion (default = TRUE). |

Value

A tibble containing the requested dataset.

Examples

```
# Examples that hit the live NYC Open Data API are guarded so CRAN checks
# do not fail when the network is unavailable or slow.
if (interactive() && curl::has_internet()) {
  endpoint <- "https://data.cityofnewyork.us/resource/erm2-nwe9.json"
  out <- try(nyc_any_dataset(endpoint, limit = 3), silent = TRUE)
  if (!inherits(out, "try-error")) {
    head(out)
  }
}
```

| | |
|-------------------|---|
| nyc_list_datasets | <i>List datasets available in nycOpenData</i> |
|-------------------|---|

Description

Retrieves the current NYC Open Data catalog and returns datasets available for use with ‘nyc_pull_dataset()’.

Usage

```
nyc_list_datasets()
```

Details

Keys are generated from dataset names using ‘janitor::make_clean_names()’.

Value

A tibble of available datasets, including generated ‘key’, dataset ‘uid’, and dataset ‘name’.

Examples

```
if (interactive() && curl::has_internet()) {
  nyc_list_datasets()
}
```

| | |
|------------------|--|
| nyc_pull_dataset | <i>Pull a NYC Open Data dataset from the NYC Open Data catalog</i> |
|------------------|--|

Description

Uses a dataset ‘key’ or ‘uid’ from ‘nyc_list_datasets()’ to pull data from NYC Open Data.

Usage

```
nyc_pull_dataset(
  dataset,
  limit = 10000,
  filters = list(),
  date = NULL,
  from = NULL,
  to = NULL,
  date_field = NULL,
  where = NULL,
  order = NULL,
  timeout_sec = 30,
  clean_names = TRUE,
  coerce_types = TRUE
)
```

Arguments

| | |
|--------------|--|
| dataset | A dataset key or UID from 'nyc_list_datasets()'. |
| limit | Number of rows to retrieve (default = 10,000). |
| filters | Optional named list of filters. Supports vectors (translated to IN()). |
| date | Optional single date (matches all times that day) using 'date_field'. |
| from | Optional start date (inclusive) using 'date_field'. |
| to | Optional end date (exclusive) using 'date_field'. |
| date_field | Optional date/datetime column to use with 'date', 'from', or 'to'. Must be supplied when 'date', 'from', or 'to' are used. |
| where | Optional raw SoQL WHERE clause. If 'date', 'from', or 'to' are provided, their conditions are AND-ed with this. |
| order | Optional SoQL ORDER BY clause. |
| timeout_sec | Request timeout in seconds (default = 30). |
| clean_names | Logical; if TRUE, convert column names to snake_case (default = TRUE). |
| coerce_types | Logical; if TRUE, attempt light type coercion (default = TRUE). |

Details

Dataset keys are generated from dataset names using 'janitor::make_clean_names()'. Because keys are derived from live catalog metadata, dataset UIDs are the more stable option.

Value

A tibble.

Examples

```
if (interactive() && curl::has_internet()) {  
  # Pull by key  
  nyc_pull_dataset("311_service_requests", limit = 3)  
  
  # Pull by UID  
  nyc_pull_dataset("erm2-nwe9", limit = 3)  
  
  # Filters  
  nyc_pull_dataset("erm2-nwe9", limit = 3, filters = list(borough = "QUEENS"))  
  
  # Date filtering  
  nyc_pull_dataset(  
    "erm2-nwe9",  
    from = "2023-01-01",  
    to = "2024-01-01",  
    date_field = "created_date",  
    limit = 100  
  )  
}
```

Index

nyc_any_dataset, 2
nyc_list_datasets, 3
nyc_pull_dataset, 3