

# Package: RSocrata (via r-universe)

September 13, 2024

**Type** Package

**Title** Download or Upload 'Socrata' Data Sets

**Description** Provides easier interaction with 'Socrata' open data portals <<https://dev.socrata.com>>. Users can provide a 'Socrata' data set resource URL, or a 'Socrata' Open Data API (SODA) web query, or a 'Socrata' ``human-friendly'' URL, returns an R data frame. Converts dates to 'POSIX' format and manages throttling by 'Socrata'. Users can upload data to 'Socrata' portals directly from R.

**Version** 1.7.15-1

**Date** 2023-08-13

**Author** Hugh Devlin, Ph. D., Tom Schenk, Jr., Gene Leynes, Nick Lucius, John Malc, Mark Silverberg, and Peter Schmeideskamp

**Maintainer** ``Gene Leynes'' <[developers@cityofchicago.org](mailto:developers@cityofchicago.org)>

**Depends** R (>= 3.3.0)

**Imports** httr (>= 1.0.0), jsonlite (>= 0.9.16), mime (>= 0.3), plyr (>= 1.8.4)

**Suggests** testthat (>= 0.10.0), roxygen2 (>= 4.1.0), readr (>= 0.2.2)

**License** MIT + file LICENSE

**URL** <https://github.com/Chicago/RSocrata>

**BugReports** <https://github.com/Chicago/RSocrata/issues>

**RoxygenNote** 7.2.3

**Encoding** UTF-8

**Repository** <https://chicago.r-universe.dev>

**RemoteUrl** <https://github.com/chicago/rsocrata>

**RemoteRef** HEAD

**RemoteSha** 7c35f916590f6d030ae133b25bb223cab495891c

## Contents

fieldName . . . . .	2
isFourByFour . . . . .	3
ls.socrata . . . . .	3
posixify . . . . .	4
read.socrata . . . . .	5
validateUrl . . . . .	6
write.socrata . . . . .	7
<b>Index</b>	<b>9</b>

---

fieldName	<i>Convert Socrata human-readable column name to field name</i>
-----------	---

---

## Description

Convert Socrata human-readable column name, as it might appear in the first row of data, to field name as it might appear in the HTTP header; that is, lower case, periods replaced with underscores#'

## Usage

```
fieldName(humanName)
```

## Arguments

humanName - a Socrata human-readable column name

## Value

Socrata field name

## Author(s)

Hugh J. Devlin, Ph. D. <Hugh.Devlin@cityofchicago.org>

## Examples

```
fieldName("Number.of.Stations") # number_of_stations
```

---

isFourByFour	<i>Checks the validity of the syntax for a potential Socrata dataset Unique Identifier, also known as a 4x4.</i>
--------------	--

---

**Description**

Will check the validity of a potential dataset unique identifier supported by Socrata. It will provide an exception if the syntax does not align to Socrata unique identifiers. It only checks for the validity of the syntax, but does not check if it actually exists.

**Usage**

```
isFourByFour(fourByFour)
```

**Arguments**

fourByFour - a string; character vector of length one

**Value**

TRUE if is valid Socrata unique identifier, FALSE otherwise

**Author(s)**

Tom Schenk Jr <tom.schenk@cityofchicago.org>

---

ls.socrata	<i>List datasets available from a Socrata domain</i>
------------	--

---

**Description**

List datasets available from a Socrata domain

**Usage**

```
ls.socrata(url)
```

**Arguments**

url - A Socrata URL. This simply points to the site root.

**Value**

an R data frame containing a listing of datasets along with various metadata.

**Author(s)**

Peter Schmiedeskamp <pschmied@uw.edu>

**Examples**

```
## Not run:  
# Download list of data sets  
df <- ls.socrata("https://soda.demo.socrata.com")  
# Check schema definition for metadata  
attributes(df)  
  
## End(Not run)
```

---

posixify

*Convert Socrata calendar\_date string to POSIX*

---

**Description**

Convert Socrata calendar\_date string to POSIX

**Usage**

```
posixify(x)
```

**Arguments**

x - character vector in one of two Socrata calendar\_date formats

**Value**

a POSIX date

**Author(s)**

Hugh J. Devlin, Ph. D. <Hugh.Devlin@cityofchicago.org>

---

read.socrata	<i>Get a full Socrata data set as an R data frame</i>
--------------	---

---

## Description

Manages throttling and POSIX date-time conversions

## Usage

```
read.socrata(  
  url,  
  app_token = NULL,  
  email = NULL,  
  password = NULL,  
  stringsAsFactors = FALSE  
)
```

## Arguments

url	- A Socrata resource URL, or a Socrata "human-friendly" URL, or Socrata Open Data Application Program Interface (SODA) query requesting a comma-separated download format (.csv suffix), May include SoQL parameters, but is assumed to not include a SODA offset parameter
app_token	- a string; SODA API token used to query the data portal <a href="https://dev.socrata.com/consumers/getting-started.html">https://dev.socrata.com/consumers/getting-started.html</a>
email	- Optional. The email to the Socrata account with read access to the dataset
password	- Optional. The password associated with the email to the Socrata account
stringsAsFactors	- Optional. Should character columns be converted to factor (TRUE or FALSE)?

## Value

an R data frame with POSIX dates

## Author(s)

Hugh J. Devlin, Ph. D. <Hugh.Devlin@cityofchicago.org>

## Examples

```
## Not run:  
# Human-readable URL:  
url <- "https://soda.demo.socrata.com/dataset/USGS-Earthquakes-for-2012-11-01-API/4334-bgaj"  
df <- read.socrata(url)  
# SoDA URL:  
df <- read.socrata("https://soda.demo.socrata.com/resource/4334-bgaj.csv")  
# Download private dataset
```

```
socrataEmail <- Sys.getenv("SOCRATA_EMAIL", "mark.silverberg+soda.demo@socrata.com")
socrataPassword <- Sys.getenv("SOCRATA_PASSWORD", "7vFDsGFDUG")
privateResourceToReadCsvUrl <- "https://soda.demo.socrata.com/resource/a9g2-feh2.csv" # dataset
read.socrata(url = privateResourceToReadCsvUrl, email = socrataEmail, password = socrataPassword)
# Using an API key to read datasets (reduces throttling)
token <- "ew2rEMuESuzWPqMkyPFOSGJgE"
df <- read.socrata("https://soda.demo.socrata.com/resource/4334-bgaj.csv",
                  app_token = token)

nrow(df)
closeAllConnections()

## End(Not run)
```

---

validateUrl	<i>Convert, if necessary, URL to valid REST API URL supported by Socrata.</i>
-------------	---

---

### Description

Will convert a human-readable URL to a valid REST API call supported by Socrata. It will accept a valid API URL if provided by users and will also convert a human-readable URL to a valid API URL. Will accept queries with optional API token as a separate argument or will also accept API token in the URL query. Will resolve conflicting API token by deferring to original URL.

### Usage

```
validateUrl(url, app_token)
```

### Arguments

url	- a string; character vector of length one
app_token	- a string; SODA API token used to query the data portal <a href="https://dev.socrata.com/consumers/getting-started.html">https://dev.socrata.com/consumers/getting-started.html</a>

### Value

a - valid Url

### Author(s)

Tom Schenk Jr <tom.schenk@cityofchicago.org>

---

write.socrata	<i>Write to a Socrata dataset (full replace or upsert)</i>
---------------	--

---

## Description

Method for updating Socrata datasets

## Usage

```
write.socrata(  
  dataframe,  
  dataset_json_endpoint,  
  update_mode,  
  email,  
  password,  
  app_token = NULL  
)
```

## Arguments

dataframe	- dataframe to upload to Socrata
dataset_json_endpoint	- Socrata Open Data Application Program Interface (SODA) endpoint (JSON only for now)
update_mode	- "UPSERT" or "REPLACE"; consult <a href="https://dev.socrata.com/publishers/getting-started.html">https://dev.socrata.com/publishers/getting-started.html</a>
email	- The email to the Socrata account with read access to the dataset
password	- The password associated with the email to the Socrata account
app_token	- a (non-required) string; SODA API token can be used to query the data portal <a href="https://dev.socrata.com/consumers/getting-started.html">https://dev.socrata.com/consumers/getting-started.html</a>

## Author(s)

Mark Silverberg <[mark.silverberg@socrata.com](mailto:mark.silverberg@socrata.com)>

## Examples

```
## Not run:  
# Store user email and password  
socrataEmail <- Sys.getenv("SOCRATA_EMAIL", "mark.silverberg+soda.demo@socrata.com")  
socrataPassword <- Sys.getenv("SOCRATA_PASSWORD", "7vFDsGFDUG")  
  
datasetToAddToUrl <- "https://soda.demo.socrata.com/resource/xh6g-yugi.json" # dataset  
  
# Generate some data  
x <- sample(-1000:1000, 1)  
y <- sample(-1000:1000, 1)
```

```
df_in <- data.frame(x,y)

# Upload to Socrata
write.socrata(df_in,datasetToAddToUrl,"UPSERT",socrataEmail,socrataPassword)

## End(Not run)
```



# Index

`fieldName`, [2](#)

`isFourByFour`, [3](#)

`ls.socrata`, [3](#)

`posixify`, [4](#)

`read.socrata`, [5](#)

`validateUrl`, [6](#)

`write.socrata`, [7](#)