

# Package ‘RPublica’

July 2, 2014

**Type** Package

**Title** ProPublica API Client

**Version** 0.1.1

**Date** 2014-05-09

**Author**

Authors@R: c(person("Thomas J.", "Leeper", role = c("aut", "cre"), email = "thosjleeper@gmail.com"))

**Maintainer** Thomas J. Leeper <thosjleeper@gmail.com>

**Imports** XML, RJSONIO, RCurl

**Description** Client for accessing ProPublica's data journalism APIs

**License** GPL-2

**URL** <https://github.com/rOpenGov/RPublica>

**BugReports** <https://github.com/rOpenGov/RPublica/issues>

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2014-05-10 00:11:40

## R topics documented:

RPublica-package . . . . .	2
committee . . . . .	2
filing . . . . .	3
forensics . . . . .	3
market . . . . .	4
nonprofitexplorer . . . . .	5
ppQuery . . . . .	6
station . . . . .	6
<b>Index</b>	<b>7</b>

RPublica-package

*ProPublica API Client*

---

**Description**

Client for accessing ProPublica's data journalism APIs.

**Author(s)**

Thomas J. Leeper

Maintainer: Thomas J. Leeper <thosjleeper@gmail.com>

**References**

APIs:

[Nonprofit Explorer](#)

[Forensics](#)

[Free the Files](#)

---

committee*Retrieve committees or committee*

---

**Description**

Retrieve available committees, or a specific committee

**Usage**

```
committee(slug = NULL, ...)
```

**Arguments**

slug            An optional character string containing a “slug” referring to a political committee.

...             Arguments passed to [getURL](#).

**Author(s)**

Thomas J. Leeper

**Examples**

```
## Not run: committee('obama-for-america')
```

---

filing

*Retrieve filing*

---

### Description

Retrieve a specific filing, by ID

### Usage

```
filing(id, ...)
```

### Arguments

id	A character string containing a filing identification number.
...	Arguments passed to <a href="#">getURL</a> .

### Author(s)

Thomas J. Leeper

### Examples

```
## Not run: filing(51212)
```

---

forensics

*Forensics API*

---

### Description

Retrieve details about a given state or system

### Usage

```
geos(state=NULL, ...)
```

```
systems(id, ...)
```

### Arguments

state	An optional character string containing a two-letter state postal code. If NULL, results for all available states are returned.
id	The identification number for a particular medical examiner or coroner system.
...	Arguments passed to <a href="#">getURL</a> .

**Author(s)**

Thomas J. Leeper

**References**

[API Documentation](#)

**Examples**

```
## Not run: geos('ny')  
## Not run: systems(36)
```

---

market	<i>Retrieve market or markets</i>
--------	-----------------------------------

---

**Description**

Retrieve available markets, or a specific market

**Usage**

```
market(slug = NULL, ...)
```

**Arguments**

slug	An optional character string containing a “slug” referring to a media market.
...	Arguments passed to <a href="#">getURL</a> .

**Author(s)**

Thomas J. Leeper

**Examples**

```
## Not run: market('new-york')
```

---

 nonprofitexplorer      *Nonprofit Explorer*


---

**Description**

Search for and retrieve nonprofit data

**Usage**

```
np_search(q=NULL, order=NULL, sort=NULL, state=NULL, ntee=NULL, subsection=NULL, ...)
```

```
np_org(ein, ...)
```

**Arguments**

q	A character string containing a search query.
order	A character string containing a field by which to order the results. One of “name”, “city”, “state”, “c_code”, “fiscalyr”, “revenue”, “expenses”, “assets”, “liabilities”, “pct_officer_compt”. The ascending or descending sort by this parameter is controlled by sort.
sort	One of “asc” or “desc”, controlling the order of the results (according the the ordering given in order).
state	A two-letter state postal abbreviation code for a U.S. state or territory. Use “ZZ” for foreign organizations.
ntee	A numeric value (between 1 and 10) containing a <a href="#">National Taxonomy of Exempt Entities (NTEE) Major Group</a> code.
subsection	A numeric value (between 2 and 92) referring to a subsection of tax code section 501(c).
ein	An employer identification number (EIN).
...	Arguments passed to <a href="#">getURL</a> .

**Value**

A list containing the API response.

**Author(s)**

Thomas J. Leeper

**References**

[API Documentation](#)

**Examples**

```
## Not run: np_search('propublica')
## Not run: np_org(142007220)
```

---

ppQuery                      *API Query*

---

### Description

Execute an API query

### Usage

```
ppQuery(op, baseUrl, fmt='json', args = NULL, ...)
```

### Arguments

op	A character string containing an API operation, and associated arguments.
baseUrl	The base URL for conducting the API query.
fmt	One of “xml” or “json” (the default).
args	A character string containing additional URL-encoded parameters.
...	Arguments passed to <a href="#">getURL</a> .

### Author(s)

Thomas J. Leeper

---

station                      *Retrieve station*

---

### Description

Retrieve station

### Usage

```
station(callsign, ...)
```

### Arguments

callsign	A character string containing a station callsign.
...	Arguments passed to <a href="#">getURL</a> .

### Author(s)

Thomas J. Leeper

### Examples

```
## Not run: station('WEWS-TV')
```

# Index

## \*Topic **package**

RPublica-package, [2](#)

committee, [2](#)

filing, [3](#)

forensics, [3](#)

geos (forensics), [3](#)

getURL, [2-6](#)

market, [4](#)

nonprofitexplorer, [5](#)

np\_org (nonprofitexplorer), [5](#)

np\_search (nonprofitexplorer), [5](#)

ppQuery, [6](#)

RPublica (RPublica-package), [2](#)

RPublica-package, [2](#)

station, [6](#)

systems (forensics), [3](#)