

# Package ‘pageviews’

November 25, 2015

**Title** An API Client for Wikimedia Traffic Data

**Version** 0.1.1

**Date** 2015-11-25

**Author** Oliver Keyes

**Maintainer** Oliver Keyes <ironholds@gmail.com>

**Description** Pageview data from the 'Wikimedia' sites, such as 'Wikipedia' (<https://www.wikipedia.org/>), from entire projects to per-article levels of granularity, through the new RESTful API and data source ([https://wikimedia.org/api/rest\\_v1/?doc](https://wikimedia.org/api/rest_v1/?doc)).

**License** MIT + file LICENSE

**LazyData** true

**URL** <https://github.com/ironholds/pageviews>

**BugReports** <https://github.com/ironholds/pageviews>

**Suggests** testthat, knitr, rmarkdown, WikipediR, WikidataR

**Imports** jsonlite, httr

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-11-25 13:55:00

## R topics documented:

article_pageviews . . . . .	2
pageviews . . . . .	3
project_pageviews . . . . .	3
top_articles . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

article\_pageviews      *Retrieve Pageview Data for an Article*

---

### Description

retrieves the pageview data for a particular article on a project, within a provided time-range.

### Usage

```
article_pageviews(project = "en.wikipedia",
  article = "R (programming language)", platform = "all",
  user_type = "all", start = "2015100100", end = NULL, reformat = TRUE,
  ...)
```

### Arguments

project	the name of the project, structured as [language_code].[project] (see the default).
article	the article(s) you want to retrieve data for. Ideally features underscores in the title instead of spaces, but happily converts if you forget to do this.
platform	The platform the pageviews came from; one of "all", "desktop", "mobile-web" and "mobile-app". Set to "all" by default.
user_type	the type of users. One of "all", "user", "spider" or "bot". "all" by default.
start	the start YYYYMMDDHH of the range you want to cover.
end	the end YYYYMMDDHH of the range you want to cover. NULL by default, meaning that it returns 1 day of data.
reformat	Whether to reformat the results as a <a href="#">data.frame</a> or not. TRUE by default.
...	further arguments to pass to httr's GET.

### See Also

[top\\_articles](#) for the top articles per project in a given date range, and [project\\_pageviews](#) for per-project pageviews.

### Examples

```
# Basic example
r_pageviews <- article_pageviews()

# Modify the article
obama_pageviews <- article_pageviews(article = "Barack_Obama")
```

---

pageviews                      *An API client for Wikimedia traffic data*

---

### Description

Pageview data from the 'Wikimedia' sites, such as Wikipedia (<https://www.wikipedia.org/>), from entire projects to by-article levels of granularity.

---

project\_pageviews            *Retrieve Per-Project Pageview Counts*

---

### Description

Retrieve pageview counts for a particular project.

### Usage

```
project_pageviews(project = "en.wikipedia", platform = "all",
  user_type = "all", granularity = "daily", start = "2015100100",
  end = NULL, reformat = TRUE, ...)
```

### Arguments

project	the name of the project, structured as [language_code].[project] (see the default).
platform	The platform the pageviews came from; one of "all", "desktop", "mobile-web" and "mobile-app". Set to "all" by default.
user_type	the type of users. One of "all", "user", "spider" or "bot". "all" by default.
granularity	the granularity of data to return; do you want hourly or daily counts? Set to "daily" by default.
start	the start YYYYMMDDHH of the range you want to cover.
end	the end YYYYMMDDHH of the range you want to cover. NULL by default, meaning that it returns 1 day/hour of data (depending on the value passed to granularity).
reformat	Whether to reformat the results as a <a href="#">data.frame</a> or not. TRUE by default.
...	further arguments to pass to http's GET.

### See Also

[top\\_articles](#) for the top articles per project in a given date range, and [article\\_pageviews](#) for per-article pageviews.

**Examples**

```
# Basic call
enwiki_1_october_pageviews <- project_pageviews()

# Break it down to hourly
enwiki_hourly <- project_pageviews(granularity = "hourly", end = "201510123")
```

---

top_articles	<i>Retrieve Data on Top Articles</i>
--------------	--------------------------------------

---

**Description**

top\_articles grabs data on the top articles for a project in a given time period, and for a particular platform.

**Usage**

```
top_articles(project = "en.wikipedia", platform = "all", year = "2015",
             month = "10", day = "01", reformat = TRUE, ...)
```

**Arguments**

project	the name of the project, structured as [language_code].[project] (see the default).
platform	The platform the pageviews came from; one of "all", "desktop", "mobile-web" and "mobile-app". Set to "all" by default.
year	The year the articles were "top" in. 2015 by default.
month	The month the articles were "top" in. "10" by default; can be set to "all", for all the months in year. If so, day must also be "all".
day	The day the articles were "top" in. "01" by default; can be set to "all", for all the days in month.
reformat	Whether to reformat the results as a <a href="#">data.frame</a> or not. TRUE by default.
...	further arguments to pass to httr's GET.

**See Also**

[article\\_pageviews](#) for per-article pageviews and [project\\_pageviews](#) for per-project pageviews.

**Examples**

```
# Basic example
enwiki_top_articles <- top_articles()

# Use a narrower platform
enwiki_mobile_top <- top_articles(platform = "mobile-web")
```

# Index

`article_pageviews`, [2](#), [3](#), [4](#)

`data.frame`, [2-4](#)

`pageviews`, [3](#)

`pageviews-package (pageviews)`, [3](#)

`project_pageviews`, [2](#), [3](#), [4](#)

`top_articles`, [2](#), [3](#), [4](#)