Package 'diezeit'

October 13, 2022

Type Package

Title R Interface to the ZEIT ONLINE Content API

Version 0.1-0

Date 2015-10-19

Author Christian Graul

Maintainer Christian Graul <christian.graul@gmail.com>

Description A wrapper for the ZEIT ONLINE Content API, available at <http://developer.zeit.de>. 'diezeit' gives access to articles and corresponding metadata from the ZEIT archive and from ZEIT ONLINE. A personal API key is required for usage.

License MIT + file LICENSE

VignetteBuilder knitr

Imports brew, grDevices, httr, jsonlite, methods, utils

Suggests knitr

NeedsCompilation no

Repository CRAN

Date/Publication 2015-10-19 20:48:03

R topics documented:

---

changes   *View changes notes.*

Description

changes brings up the NEWS file of the package.

Usage

```
changes(pkg = "diezeit")
```

Arguments

pkg   Set to the default "diezeit". Other packages make no sense.

Examples

```
## Not run:
changes()

## End(Not run)
```

---

diezeit   *ZEIT ONLINE Content API*

Description

A wrapper for the ZEIT ONLINE Content API, available at <http://developer.zeit.de>. It gives access to articles and corresponding metadata from the ZEIT archive and from ZEIT ONLINE. A personal API key is required for usage.

Details

Accessing the ZEIT archive requires an API key, that can be requested at http://developer.zeit.de/quickstart. Registration is free and allows for API-Access with a limit of 10,000 requests per day. If you do not want to enter your key for each R session, put the following in your .Renviron or .Rprofile file: ZEIT_KEY=PUTYOURKEYHERE

See Also

zeit_client for client information and usage, zeit_search for ZEIT archive search or zeit_get to get content from the ZEIT archive.

---

zeit_client *Observe your usage*

Description

zeit_client does not provide content per se, but lets you get information about your API usage.

Usage

```
zeit_client(print = TRUE)
```

Arguments

print   if TRUE (default), the client information is printed.

Value

a list of information about the client and API usage

Examples

```
## Not run:
zeit_client()

## End(Not run)
```

---

zeit_get *Get detailled content from the ZEIT archive*

Description

zeit_get will get you all available metadata for a specific item.

Usage

```
zeit_get(endpoint, id, fields, print = TRUE)
```

Arguments

endpoint   one of author, content, department, keyword, product or series – see
             [zeit_search](#).

id   item id.

fields   partially select output fields, as string value or vector of strings for multiple
             fields.

print   if TRUE (default) the meta data are printed.

Details

Endpoints

The API is structured into several endpoints that provide specific functionalities:

author	content by this author
content	get content by ID
department	content from this department
keyword	content about this keyword
product	content from this product
series	content in this series

Value

List of metadata items.

Source

<http://developer.zeit.de/docs/>

Examples

```
## Not run:
# get article metadata by ID
zeit_get("content", "3Ed7KYJOO2MXu5SQtnudQA")

# partial selection of output fields
zeit_get("content", "3Ed7KYJOO2MXu5SQtnudQA",
  fields=c("title", "release_date", "href"))

# hide result
article.meta <- zeit_get("content", "3Ed7KYJOO2MXu5SQtnudQA", print=FALSE)

## End(Not run)
```

---

zeit_search   *Search the ZEIT archive*

Description

zeit_search exposes a search for ZEIT archive items. You can set search queries, paginate, sort and partially select the fields, that should be returned. Articles, that match your query, are returned with a reduced set of meta data.

Usage

```
zeit_search(endpoint, query, fields, limit = 10, offset = 0, sort,
  print = TRUE)
```

Arguments

endpoint	one of `author`, `content`, `department`, `keyword`, `product` or `series` – provides specific search functionalities.
query	the main search query; single string value or vector of strings.
fields	partially select output fields, as string value or vector of strings for multiple fields.
limit	limit the amount of matches to return; set to `10` by default.
offset	offset for the list of matches; set to `0` by default.
sort	sort search results by any of the returned `fields`. Vector of two (`c([field], [direction])`), giving field and direction keyword. Direction keywords are `asc` and `desc` for an ascending or descending sort order respectively. Multiple sort orders are accepted as `list` of such vectors.
print	if `TRUE` (default) the search results are printed.

Details

Endpoints

The API is structured into several endpoints that provide specific functionalities:

author	search all authors
content	search for content
department	search all departments
keyword	search all keywords
product	search all products
series	search all series

Query syntax

You can search the entire article text and all meta data simply by setting the query parameter to your search phrase. The search uses entire strings "as is". To search for multiple tokens use a vector of strings.

All fields of an article can be queried individually by using `[field]:[search string]`. For example, to get articles that have the word "Kennedy" in their headline, you would search for `"title:Kennedy"`.

Currently all `endpoints` other than `content` only support simple search phrases with asterisk (`*`) wildcards.

Value

A list of matches to the query.

Source

http://developer.zeit.de/docs/

Examples

```
## Not run:
# simple content search
zeit_search(endpoint="content", query="bayreuth")
zeit_search("content", "bayreuth") # same same

# multiple tokens
zeit_search("content", c("bayreuth", "festspiele"))

# entire string
zeit_search("content", "bayreuther festspiele")

# field query
zeit_search("content", "title:bayreuth")

# partial selection
zeit_search("content", "bayreuth", fields=c("title", "teaser_text"))

# pagination
zeit_search("content", "bayreuth", limit=1) # just one match
zeit_search("content", "bayreuth", limit=1, offset=1) # just the second match

# sorting
zeit_search("content", "bayreuth",
  sort=c("release_date", "asc")) # sort by date
zeit_search("content", "bayreuth",
  sort=list(c("release_date", "desc"), c("title", "asc"))) # sort by date and title

# hide matches
bt.matches <- zeit_search("content", "bayreuth", print=FALSE)

# author search
zeit_search(endpoint="author", query="Stefan Locke")

## End(Not run)
```

Index