

# Package ‘europeanaR’

May 17, 2022

**Type** Package

**Title** Interact with Metadata Records and Media on the Europeana Repository

**Version** 0.1.0

**Maintainer** Alexandros Kouretsis <akourets@ionio.gr>

**Description** Interact with the Europeana Data Model via a variety of API endpoints that contains digital collections from thousands of institutions around Europe. This translates to millions of Cultural Heritage Objects in the form of image, text, video, sound and 3D, accompanied by rich metadata. The Data Model design principles are based on the core principles and best practices of the Semantic Web and Linked Data efforts to which Europeana contributes (see, e.g., Doerr, Martin, et al. The europeana data model (edm). World Library and Information Congress: 76th IFLA general conference and assembly. Vol. 10. 2010.). The package also provides methods for bulk downloads of specific subsets of items, including both their metadata and their associated media files.

**License** MIT + file LICENSE

**Encoding** UTF-8

**URL** <https://github.com/AleKoure/europeanaR>

**Imports** data.table, httr, jsonlite, magrittr, utils, Rdpack

**RdMacros** Rdpack

**RoxygenNote** 7.1.2

**Suggests** covr, knitr, rmarkdown, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**VignetteBuilder** knitr

**NeedsCompilation** no

**Author** Alexandros Kouretsis [aut, cre],  
Andreas Giannakouloupoulos [aut],  
Laida Limniati [aut]

**Repository** CRAN

**Date/Publication** 2022-05-17 16:40:02 UTC

## R topics documented:

download_media	2
get_key	3
query_record_api	3
query_search_api	4
set_key	6
tidy_cursored_search	7
tidy_search_items	8

<b>Index</b>	<b>9</b>
--------------	----------

---

download_media	<i>Download Media</i>
----------------	-----------------------

---

### Description

Function that downloads media, using the response object of the Europeana search API. It uses the fields 'type' and 'edmIsShownBy' to retrieve the items and store them in a local folder.

### Usage

```
download_media(resp, download_dir = NULL, type_ = NULL, quiet = TRUE)
```

### Arguments

resp	an S3 object of type 'europeana_search_api' or 'cursored_search'
download_dir	destination directory. If 'NULL' then 'tempdir()' is used
type_	string in 'c("TEXT", "IMAGE", "SOUND", "VIDEO", "3D")'
quiet	boolean to suppress download file messages

### Value

destination folder

### Examples

```
#set your API key with set_key(api_key = "XXXX")
#example_1
resp <- query_search_api("arioch", rows = 2)
download_media(resp, type = "IMAGE")

#example_2 bulk download
res_bulk <- tidy_cursored_search(query = "animal",
                                qf = "when:17 AND what:painting",
                                max_items = 3)

download_media(res_bulk)
```

---

get_key	<i>Get API key</i>
---------	--------------------

---

**Description**

This function is a simple wrapper of the ‘Sys.getenv’ base function. It gets the value of the environmental variable ‘EUROPEANA\_KEY’.

**Usage**

```
get_key()
```

**Value**

character with the API key stored as environmental variable

**See Also**

[set\\_key](#)

---

query_record_api	<i>Query Record API</i>
------------------	-------------------------

---

**Description**

The Record API provides direct access to the Europeana data, which is modeled using the Europeana Data Model (EDM). While EDM is an open flexible data model featuring various kind of resources and relations between them, the Record API (and the Europeana Collections Portal) supports the retrieval of a segment of EDM for practical purposes.

These "atomic" EDM segments typically contain one Cultural Heritage Object (CHO), aggregation information that connects the metadata and digital representations, and a number of contextual resources related to the CHO, such as agents, locations, concepts, and time.

**Usage**

```
query_record_api(id, path = "/record/v2", ...)
```

**Arguments**

id	string with the ‘RECORD_ID’ in the form of ‘/DATASET_ID/LOCAL_ID’
path	string that indicates version of the API
...	other parameters passed as query parameters

**Value**

S3 object of class ‘europeana\_record\_api’. Contains the parsed content, the path, and the API response compatible with ‘httr’ methods.

**Source**

<https://pro.europeana.eu/page/record>

**References**

Doerr M, Gradmann S, Hennicke S, Isaac A, Meghini C, Van de Sompel H (2010). “The europeana data model (edm).” In *World Library and Information Congress: 76th IFLA general conference and assembly*, volume 10, 15.

Wickham H (2020). *httr: Tools for Working with URLs and HTTP*. <https://httr.r-lib.org/>, <https://github.com/r-lib/httr>.

Ooms J (2014). “The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects.” *arXiv:1403.2805 [stat.CO]*. <https://arxiv.org/abs/1403.2805>.

**Examples**

```
#set your API key with set_key(api_key = "XXXX")
#query search API
res <- query_search_api("arioch", qf = "1712", media = TRUE)
#get results in tidy format
dat <- tidy_search_items(res)
#query records API for each item
lapply(dat$id, query_record_api)
```

---

query\_search\_api

*Europeana Search API*

---

**Description**

The Search API allows to search the Europeana repository for metadata records and media. The Search API is the most straightforward to use. It works in a similar fashion to the Europeana website when it comes to interacting with the data. You can use the API to search for keywords, and it will return any entries that contain those words. You can refine your search with more advanced queries like Boolean Searches, or you can filter out parts of the results advanced filtering.

**Usage**

```

query_search_api(
  query = NULL,
  rows = NULL,
  profile = NULL,
  qf = NULL,
  reusability = NULL,
  media = NULL,
  thumbnail = NULL,
  landingpage = NULL,
  colourpalette = NULL,
  facet = NULL,
  limit = NULL,
  start = NULL,
  sort = NULL,
  path = "/record/v2/search.json",
  ...
)

```

**Arguments**

query	(character) string with the search term(s)
rows	(numeric) that indicates the number of records to return
profile	(character) Profile parameter controls the format and richness of the response.
qf	(character) Facet filtering query. This parameter can be defined more than once.
reusability	(character) Filter by copyright status.
media	(character) Filter by records where an URL to the full media file is present in the edm:isShownBy or edm:hasView metadata and is resolvable.
thumbnail	(character) Filter by records where a thumbnail image has been generated for any of the WebResource media resources (thumbnail available in the edmPreview field).
landingpage	(character) Filter by records where the link to the original object on the providers website (edm:isShownAt) is present and verified to be working.
colourpalette	(character) Filter by images where one of the colours of an image matches the provided colour code. You can provide this parameter multiple times, the search will then do an 'AND' search on all the provided colours. See colour palette.
facet	(character) Name of an individual facet. See individual facets.
limit	(numeric) The number of records to return. Maximum is 100. Default: 10
start	(numeric) The item in the search results to start with. The first item is 1. Default: 1
sort	(character) Sort by a field, e.g., timestamp_update+desc
path	(character) URL signature with the API version
...	parameters passed in get request

## Details

In the ‘query’ parameter the Apache Lucene Query Syntax is inherently supported by the Search API. Users can use Lucene and Apache SOLR guides to get the most out of the Europeana repository.

The response is always formatted in JSON and will contain a number of fields that present information about the handling of the request, while the concrete information about the record is presented in the "items" field (see Metadata Sets). The parsed information is stored in the ‘content’ field of the S3 object returned.

## Value

S3 object of class ‘europeana\_search\_api’. Contains the parsed content, the path, and the API response compatible with ‘httr’ methods.

## Source

<https://pro.europeana.eu/page/search>

## References

Doerr M, Gradmann S, Hennicke S, Isaac A, Meghini C, Van de Sompel H (2010). “The europeana data model (edm).” In *World Library and Information Congress: 76th IFLA general conference and assembly*, volume 10, 15.

Wickham H (2020). *httr: Tools for Working with URLs and HTTP*. <https://httr.r-lib.org/>, <https://github.com/r-lib/httr>.

Ooms J (2014). “The jsonlite Package: A Practical and Consistent Mapping Between JSON Data and R Objects.” *arXiv:1403.2805 [stat.CO]*. <https://arxiv.org/abs/1403.2805>.

## Examples

```
#set your API key with set_key(api_key = "XXXX")
#query search API
res <- query_search_api("arioch", qf = "1712", media = TRUE)
```

---

set\_key

*Set API key*

---

## Description

This function is a simple wrapper of the ‘Sys.setenv’ base function. It sets the value of the environmental variable ‘EUROPEANA\_KEY’. Alternatively, use .Renviron to set the key. Get and API key in the following link <https://pro.europeana.eu/page/get-api>.

**Usage**

```
set_key(api_key)
```

**Arguments**

api\_key            string with the API key

**Value**

No return value, called for setting the environmental variable 'EUROPEANA\_KEY'.

**See Also**

[setkey](#)

---

tidy\_cursored\_search    *Tidy Cursored Search*

---

**Description**

This function is a "runner" of a particular query that consequently makes API requests until maximum items are reached or all related items have been collected.

**Usage**

```
tidy_cursored_search(query, max_items = 10000, ...)
```

**Arguments**

query            string with the search term(s)  
max\_items        numeric that indicates max items collected  
...              params passed to get request, see also 'query\_search\_api'

**Value**

S3 object of type 'cursored\_search'. Contains a 'data.table' with all the responses transformed to tabular format, the path to the first request that starts the cursored search, and the corresponding response object compatible with 'httr' methods.

## Examples

```
#set your API key with set_key(api_key = "XXXX")
#query search API up to 3 items
res <- tidy_cursored_search(query = "animal",
                           max_items = 3,
                           theme = "art",
                           media = TRUE)

head(res$data[, 1:3])
```

---

tidy_search_items	<i>Tidy search API response</i>
-------------------	---------------------------------

---

## Description

Transforms API response to a tidy ‘data.table’ for easier manipulation

## Usage

```
tidy_search_items(resp)
```

## Arguments

resp                    an S3 object of type ‘europeana\_search\_api’

## Value

‘data.table’ with stacked results collected from the search api. Each row corresponds to a Cultural Heritage Object.

## Examples

```
#set your API key with set_key(api_key = "XXXX")
resp <- query_search_api("arioch")
tidy_search_items(resp)
```



# Index

`download_media`, [2](#)

`get_key`, [3](#)

`query_record_api`, [3](#)

`query_search_api`, [4](#)

`set_key`, [3](#), [6](#)

`setkey`, [7](#)

`tidy_cursored_search`, [7](#)

`tidy_search_items`, [8](#)