

Package ‘repec’

July 23, 2025

Type Package

Title Access RePEc Data Through API

Version 0.1.0

Description Utilities for accessing RePEc (Research Papers in Economics) through a RESTful API. You can request a code and get detailed information at the following page: <<https://ideas.repec.org/api.html>>.

License MIT + file LICENSE

URL <https://github.com/chrMongeau/repec>

BugReports <https://github.com/chrMongeau/repec/issues>

Imports jsonlite, utils

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

NeedsCompilation no

Author Christian Alexander Mongeau Ospina [aut, cre] (ORCID: <<https://orcid.org/0000-0002-4047-3924>>)

Maintainer Christian Alexander Mongeau Ospina <christian+rpkg-repec@mongeau.net>

Repository CRAN

Date/Publication 2018-08-31 18:20:08 UTC

Contents

repec-package	2
areyouthere	3
getauthornep	3
getauthorrecordfull	4
getauthorrecordraw	5
getauthorsforitem	5
getauthorshortid	6
getauthorstats	7

getauthortwitter	7
getfirstpubyear	8
getgenealogy	9
gethindex	9
getinstrecord	10
getjelforitem	11
repec_api_with_id	11
testip	12
whatismyip	13
Index	14

repec-package	<i>repec: access RePEc data through API</i>
---------------	---

Description

repec provides a way to interact with the RePEc API (Application Programming Interface) in order to get various information stored on RePEc about authors, items, institutions, etc.

Details

By using the functions of this package, you will be able to import RePEc information in R. You will need to ask for an access code by following the instructions at <https://ideas.repec.org/api.html>. The code will be needed when calling most of the functions by using the code parameter. One way to avoid passing explicitly the code is to create the REPEC_API_KEY environment variable. The code below can be put in your .RProfile so that it is always available when you start R:

```
# YOURCODE is your personal access code
Sys.setenv(REPEC_API_KEY = 'YOURCODE')
```

Author(s)

Maintainer: Christian Alexander Mongeau Ospina <christian+rpkg-repec@mongeau.net> (0000-0002-4047-3924)

See Also

Useful links:

- <https://github.com/chrMongeau/repec>
- Report bugs at <https://github.com/chrMongeau/repec/issues>

areyouthere	<i>Verifies whether the server is alive</i>
-------------	---

Description

Verifies whether the server is alive

Usage

```
areyouthere()  
  
are_you_there()
```

Value

TRUE if the server if alive, FALSE otherwise.

Examples

```
## Not run:  
areyouthere()  
  
## End(Not run)
```

getauthornep	<i>Get the NEP fields for the papers of an author</i>
--------------	---

Description

Get the NEP fields for the papers of an author

Usage

```
getauthornep(id, code = NA)  
  
get_author_nep(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame

Examples

```
## Not run:  
getauthornep('pmo915', code = YOURCODE)  
  
## End(Not run)
```

getauthorrecordfull	<i>Provides info about author, mostly in the form of handles</i>
---------------------	--

Description

Provides info about author, mostly in the form of handles

Usage

```
getauthorrecordfull(id, code = NA)  
  
get_author_record_full(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame

Examples

```
## Not run:  
getauthorrecordfull('pmo915', code = YOURCODE)  
  
## End(Not run)
```

getauthorrecordraw	<i>Provides extended info about author</i>
--------------------	--

Description

Provides extended info about author

Usage

```
getauthorrecordraw(id, code = NA)

get_author_record_raw(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame. Note: columns article and paper are data.frames.

Examples

```
## Not run:
getauthorrecordraw('pmo915', code = YOURCODE)

## End(Not run)
```

getauthorsforitem	<i>Get Authors for an item</i>
-------------------	--------------------------------

Description

Get Authors for an item

Usage

```
getauthorsforitem(id, code = NA)

get_authors_for_item(id, code = NA)
```

Arguments

id	RePEc handle for an item.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame.

Examples

```
## Not run:
getauthorsforitem('RePEc:eee:enepol:v:78:y:2015:i:c:p:41-50', code = YOURCODE)

## End(Not run)
```

getauthorshortid	<i>Get Short-ID and name from string in name or email</i>
------------------	---

Description

Get Short-ID and name from string in name or email

Usage

```
getauthorshortid(x, code = NA)

get_author_short_id(x, code = NA)
```

Arguments

x	string.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame

Examples

```
## Not run:
getauthorshortid('christian', code = YOURCODE)

## End(Not run)
```

getauthorstats	<i>Provides Author's statistics</i>
----------------	-------------------------------------

Description

Provides Author's statistics

Usage

```
getauthorstats(id, code = NA)

get_author_stats(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

Author's statistics.

Examples

```
## Not run:
getauthorstats('pmo915', code = YOURCODE)

## End(Not run)
```

getauthortwitter	<i>Provides Twitter handle</i>
------------------	--------------------------------

Description

Provides Twitter handle

Usage

```
getauthortwitter(id, code = NA)

get_author_twitter(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

String with the Twitter author linked to the Short-ID.

Examples

```
## Not run:
getauthortwitter('pmo915', code = YOURCODE)

## End(Not run)
```

getfirstpubyear	<i>Provides Author's first publication year</i>
-----------------	---

Description

Provides Author's first publication year

Usage

```
getfirstpubyear(id, code = NA)

get_first_pub_year(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

Author's first publication year.

Examples

```
## Not run:
getfirstpubyear('pmo915', code = YOURCODE)

## End(Not run)
```

getgenealogy	<i>Provides Author's genealogy</i>
--------------	------------------------------------

Description

Provides Author's genealogy

Usage

```
getgenealogy(id, code = NA)

get_genealogy(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

Author's genealogy.

Examples

```
## Not run:
getgenealogy('pmo915', code = YOURCODE)

## End(Not run)
```

gethindex	<i>Provides H-index</i>
-----------	-------------------------

Description

Provides H-index

Usage

```
gethindex(id, code = NA)

get_h_index(id, code = NA)
```

Arguments

id	Short-ID.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

H-index.

Examples

```
## Not run:
gethindex('pmo915', code = YOURCODE)

## End(Not run)
```

getinstrecord	<i>Get record of Institution</i>
---------------	----------------------------------

Description

Get record of Institution

Usage

```
getinstrecord(id, code = NA)

get_inst_record(id, code = NA)
```

Arguments

id	RePEc handle for an institution
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

data.frame.

Examples

```
## Not run:
getinstrecord('RePEc:edi:dechiit', code = YOURCODE)

## End(Not run)
```

getjelforitem	<i>Get JEL codes for item</i>
---------------	-------------------------------

Description

Get JEL codes for item

Usage

```
getjelforitem(id, code = NA)
get_jel_foritem(id, code = NA)
```

Arguments

id	RePEc handle for an item.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

Vector with JEL codes.

Examples

```
## Not run:
getjelforitem('RePEc:eee:enepol:v:78:y:2015:i:c:p:41-50', code = YOURCODE)

## End(Not run)
```

repec_api_with_id	<i>Generic API call, with an ID</i>
-------------------	-------------------------------------

Description

Generic API call, with an ID

Usage

```
repec_api_with_id(id = NA, method = NA, code = NA)
```

Arguments

id	Short-ID.
method	API method to call.
code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.

Value

String with the Twitter author linked to the Short-ID.

Examples

```
## Not run:
repec_api_with_id(id = 'pmo915',
  method = 'getauthortwitter', code = YOURCODE)

## End(Not run)
```

testip	<i>Verifies whether the connection is performed from the registered IP</i>
--------	--

Description

Verifies whether the connection is performed from the registered IP

Usage

```
testip(code = NA)

test_ip(code = NA)
```

Arguments

code	User code required to use API. Included in every API call. If not supplied, the REPEC_API_KEY environment variable will be used.
------	--

Value

TRUE if the connection is from a registered IP, FALSE otherwise.

Examples

```
## Not run:
testip(YOURCODE)

## End(Not run)
```

`whatismyip`*Returns the IP address you use to call the API*

Description

Returns the IP address you use to call the API

Usage

```
whatismyip()
```

```
what_is_myip()
```

Value

String with IP.

Examples

```
## Not run:  
whatismyip()
```

```
## End(Not run)
```

Index

`are_you_there` (`areyouthere`), [3](#)
`areyouthere`, [3](#)

`get_author_nep` (`getauthornep`), [3](#)
`get_author_record_full`
 (`getauthorrecordfull`), [4](#)
`get_author_record_raw`
 (`getauthorrecordraw`), [5](#)
`get_author_short_id` (`getauthorshortid`),
 [6](#)
`get_author_stats` (`getauthorstats`), [7](#)
`get_author_twitter` (`getauthortwitter`), [7](#)
`get_authors_for_item`
 (`getauthorsforitem`), [5](#)
`get_first_pub_year` (`getfirstpubyear`), [8](#)
`get_genealogy` (`getgenealogy`), [9](#)
`get_h_index` (`gethindex`), [9](#)
`get_inst_record` (`getinstrecord`), [10](#)
`get_jel_foritem` (`getjelforitem`), [11](#)
`getauthornep`, [3](#)
`getauthorrecordfull`, [4](#)
`getauthorrecordraw`, [5](#)
`getauthorsforitem`, [5](#)
`getauthorshortid`, [6](#)
`getauthorstats`, [7](#)
`getauthortwitter`, [7](#)
`getfirstpubyear`, [8](#)
`getgenealogy`, [9](#)
`gethindex`, [9](#)
`getinstrecord`, [10](#)
`getjelforitem`, [11](#)

`repec` (`repec-package`), [2](#)
`repec-package`, [2](#)
`repec_api_with_id`, [11](#)

`test_ip` (`testip`), [12](#)
`testip`, [12](#)

`what_is_myip` (`whatismyip`), [13](#)
`whatismyip`, [13](#)