

Package ‘geniusr’

December 6, 2017

Title Tools for Working with the 'Genius' API

Version 1.0.0

Description Provides tools to interact nicely with the 'Genius' API <<https://docs.genius.com/>>. Search hosted content, extract different types of associated metadata and retrieve lyrics with ease.

URL <https://github.com/ewenme/geniusr>

Depends R (>= 3.2.0)

License MIT + file LICENSE

Encoding UTF-8

LazyData yes

RoxygenNote 6.0.1

Imports httr, purrr, tibble, rvest, stringr, xml2

NeedsCompilation no

Author Ewen Henderson [aut, cre]

Maintainer Ewen Henderson <ewenhenderson@gmail.com>

Repository CRAN

Date/Publication 2017-12-06 09:18:44 UTC

R topics documented:

geniusr	2
genius_token	2
get_album_meta	3
get_artist_meta	3
get_artist_songs	4
get_song_meta	4
scrape_lyrics_id	5
scrape_lyrics_url	5
scrape_tracklist	6
search_artist	6
search_song	7

Index**8**

geniusr	geniusr package
---------	-----------------

Description

An interface to Genius' Web API.

Details

See the README on [GitHub](#)

genius_token	Get or set GENIUS_API_TOKEN value
--------------	-----------------------------------

Description

The API wrapper functions in this package all rely on a Genius client access token residing in the environment variable GENIUS_API_TOKEN. The easiest way to accomplish this is to set it in the ``.Renvirom`` file in your home directory.

Usage

```
genius_token(force = FALSE)
```

Arguments

force	force setting a new Genius API token for the current environment?
-------	---

Value

atomic character vector containing the Genius API token

get_album_meta	<i>Retrieve meta data for an album</i>
----------------	--

Description

The Genius API lets you search for an album's meta data, given an album ID.

Usage

```
get_album_meta(album_id, access_token = genius_token())
```

Arguments

album_id	An album ID (album_id returned in get_song_meta)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
get_album_meta(album_id = 337082)  
  
## End(Not run)
```

get_artist_meta	<i>Retrieve meta data for an artist</i>
-----------------	---

Description

The Genius API lets you search for meta data for an artist, given an artist ID.

Usage

```
get_artist_meta(artist_id, access_token = genius_token())
```

Arguments

artist_id	An artist ID (artist_id returned in search_artist)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
get_artist_meta(artist_id = 16751)  
  
## End(Not run)
```

get_artist_songs	<i>Retrieve meta data for all of an artist's appearances on Genius</i>
------------------	--

Description

Return meta data for all appearances (features optional) of an artist on Genius.

Usage

```
get_artist_songs(artist_id, include_features = FALSE,  
                 access_token = genius_token())
```

Arguments

artist_id	An artist ID (artist_id returned in search_artist)
include_features	Whether to return results where artist isn't the primary artist (logical, defaults to FALSE)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
get_artist_songs(artist_id = 1421)  
  
## End(Not run)
```

get_song_meta	<i>Retrieve meta data for a song</i>
---------------	--------------------------------------

Description

The Genius API lets you search for meta data for a song, given a song ID.

Usage

```
get_song_meta(song_id, access_token = genius_token())
```

Arguments

song_id	A song ID (song_id returned in search_song)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
get_song_meta(song_id = 3039923)  
  
## End(Not run)
```

scrape_lyrics_id	<i>Retrieve lyrics associated with a Genius song ID</i>
------------------	---

Description

Scrape lyrics from Genius' lyric pages using an associated song ID.

Usage

```
scrape_lyrics_id(song_id, access_token = genius_token())
```

Arguments

song_id	song ID (like in song_id returned by search_song)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
scrape_lyrics_id(song_id = 3214267)  
  
## End(Not run)
```

scrape_lyrics_url	<i>Retrieve lyrics associated with a Genius lyrics page URL</i>
-------------------	---

Description

Scrape lyrics from a Genius' lyric page using it's associated URL. Best used with [scrape_tracklist](#), when song IDs aren't returned - otherwise, [scrape_lyrics_id](#) is recommended.

Usage

```
scrape_lyrics_url(song_lyrics_url, access_token = genius_token())
```

Arguments

song_lyrics_url	song lyrics url (like in song_lyrics_url returned by get_song_meta)
access_token	Genius' client access token, defaults to genius_token

Examples

```
## Not run:
scrape_lyrics_url(song_lyrics_url = "https://genius.com/Kendrick-lamar-dna-lyrics")

## End(Not run)
```

scrape_tracklist *Retrieve an album's tracklisting*

Description

Scrape an album's tracklisting, and song meta data, given an album ID.

Usage

```
scrape_tracklist(album_id, access_token = genius_token())
```

Arguments

album_id An album ID (album_id returned in [get_song_meta](#))
access_token Genius' client access token, defaults to genius_token

Examples

```
## Not run:
scrape_tracklist(album_id = 337082)

## End(Not run)
```

search_artist *Retrieve artist identifiers for associated search terms*

Description

The Genius API lets you search hosted content (all songs). Use search_artist to return artist_id, artist_name and artist_url for all unique artist matches to a search.

Usage

```
search_artist(search_term, n_results = 10, access_token = genius_token())
```

Arguments

search_term A character string to search for artist matches
n_results Maximum no. of search results to return (this is the number of hosted content search results, unique artist matches will be smaller)
access_token Genius' client access token, defaults to genius_token

Examples

```
## Not run:  
search_artist(search_term = "Lil")  
  
## End(Not run)
```

search_song

Retrieve song identifiers for associated search terms

Description

The Genius API lets you search hosted content (all songs). Use `search_song` to return `song_id`, `song_name`, `lyrics_url` and `artist_id` for all unique song matches to a search.

Usage

```
search_song(search_term, n_results = 10, access_token = genius_token())
```

Arguments

<code>search_term</code>	A character string to search for song matches
<code>n_results</code>	Maximum no. of search results to return
<code>access_token</code>	Genius' client access token, defaults to <code>genius_token</code>

Examples

```
## Not run:  
search_song(search_term = "Gucci", n_results=50)  
  
## End(Not run)
```

Index

genius_token, 2
geniusr, 2
geniusr-package (geniusr), 2
get_album_meta, 3
get_artist_meta, 3
get_artist_songs, 4
get_song_meta, 3, 4, 5, 6

scrape_lyrics_id, 5, 5
scrape_lyrics_url, 5
scrape_tracklist, 5, 6
search_artist, 3, 4, 6
search_song, 4, 5, 7