

Package: rgenius (via r-universe)

September 4, 2024

Type Package

Date 2020-05-02

Title Get 'Genius' API Lyrics

Version 0.1.0

Author Alberto Almuiña <albertogonzalezalmuinha@gmail.com>

Maintainer Alberto Almuiña <albertogonzalezalmuinha@gmail.com>

Description Download the lyrics of your favorite songs in text and table formats. Also search for related songs or song information. More information: <<https://docs.genius.com/>> .

URL <https://github.com/AlbertoAlmuinha/rgenius>

BugReports <https://github.com/AlbertoAlmuinha/rgenius/issues>

Imports dplyr, purrr, stringr, httr, rvest, foreach, tibble, parallel, doParallel

License GPL-3

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Repository <https://albertoalmuinha.r-universe.dev>

RemoteUrl <https://github.com/albertoalmuinha/rgenius>

RemoteRef HEAD

RemoteSha 4c8f46b112c29e170dc745aeb793bbf57a288d31

Contents

get_discography_lyrics	2
get_genius_artist	3
get_genius_artist_songs	3
get_genius_song	4
get_genius_song_lyrics	5

get_song_media	6
get_song_producer_artist	7
get_song_relationship	8
search_genius_artist	8
search_genius_song	9

Index	11
--------------	-----------

get_discography_lyrics
Get Discography Lyrics

Description

This function gets the lyrics of the desired song.

Usage

```
get_discography_lyrics(artist_id, cores = detectCores(),
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

artist_id	Genius Artist ID
cores	Number of cores to use for parallel computing. Default: parallel::detectCores()
access_token	Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_artist' to get the artist ID. This function uses parallel processing to increase performance. Use the different cores of your device to achieve this, so there may be noticeable differences in performance depending on the device.

Value

Returns a named list containing all the lyrics of the artist's songs.

Author(s)

Alberto Almuiña

Examples

```
## Not run:
get_discography_lyrics('343657', cores = 2)

## End(Not run)
```

get_genius_artist *Get Artist*

Description

This function gets 'genius' artist information

Usage

```
get_genius_artist(artist_id,  
                  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

artist_id Genius Artist ID
access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_artist' to get the artist ID.

Value

Returns a data frame with the artist's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_genius_artist('343657')  
  
## End(Not run)
```

get_genius_artist_songs
 Get Artist Songs

Description

This function gets all the songs of the artist in genius.

Usage

```
get_genius_artist_songs(artist_id,  
    access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

artist_id Genius Artist ID
access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_artist' to get the artist ID.

Value

Returns a data frame with the artist's songs information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_genius_artist_songs('343657')  
  
## End(Not run)
```

get_genius_song	<i>Get Song</i>
-----------------	-----------------

Description

This function gets 'genius' song information

Usage

```
get_genius_song(song_id, access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_id Genius Song ID
access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_song' to get the song ID.

Value

Returns a data frame with the song's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_genius_song('3440889')  
  
## End(Not run)
```

```
get_genius_song_lyrics  
  Get Song Lyrics
```

Description

This function gets the lyrics of the desired song.

Usage

```
get_genius_song_lyrics(song_id, output = "tibble", url = NULL,  
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_id	Genius Song ID
output	'tibble' or 'text'. Default: tibble.
url	If for some reason the function cannot find the url to download the letter, you can use a genius url to download it.
access_token	Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_song' to get the song ID.

Value

Returns a tibble or a character vector with the lyrics

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_genius_song_lyrics('55882', output = 'text')  
  
## End(Not run)
```

get_song_media	<i>Get Song Media</i>
----------------	-----------------------

Description

This function gets 'genius' media information

Usage

```
get_song_media(song_id, access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_id	Genius Song ID
access_token	Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_song' to get the song ID. This function returns the provider (YouTube, Spotify ..), the type (audio, video) and a link for the content.

Value

Returns a tibble with the media's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_song_media('55882')  
  
## End(Not run)
```

get_song_producer_artist
Get Song Producer

Description

This function gets song producer information

Usage

```
get_song_producer_artist(song_id,  
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_id Genius Song ID
access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_song' to get the song ID. This function returns the producer name and links to their genius profiles and images.

Value

Returns a tibble with the producer's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_song_producer_artist('55882')  
  
## End(Not run)
```

get_song_relationship *Get Song Relationship*

Description

This function fetches songs related to the wanted one.

Usage

```
get_song_relationship(song_id,  
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_id Genius Song ID
access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

To use this function, you need to first use 'search_genius_song' to get the song ID.

Value

Returns a tibble with related song's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
get_song_relationship('55882')  
  
## End(Not run)
```

search_genius_artist *Search Artist*

Description

This function searches 'genius' artists by name

Usage

```
search_genius_artist(artist_name,  
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```


Arguments

artist_name String of artist name
 access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

All Genius API endpoints require an artist or song ID. The 'search_' functions in this package try to search for the desired artists/songs to obtain their Genius ID and to use them later.

Value

Returns a data frame with the artist's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:
search_genius_artist('Arctic Monkeys')

## End(Not run)
```

search_genius_song *Search Song*

Description

This function searches 'genius' songs by name

Usage

```
search_genius_song(song_name,
  access_token = Sys.getenv("GENIUS_API_TOKEN"))
```

Arguments

song_name String of song name
 access_token Genius Web API token. Defaults to Sys.getenv('GENIUS_API_TOKEN').

Details

All Genius API endpoints require an artist or song ID. The 'search_' functions in this package try to search for the desired artists/songs to obtain their Genius ID and to use them later.

Value

Returns a data frame with the song's information

Author(s)

Alberto Almuiña

Examples

```
## Not run:  
search_genius_song('Paradise')  
  
## End(Not run)
```

Index

[get_discography_lyrics](#), 2
[get_genius_artist](#), 3
[get_genius_artist_songs](#), 3
[get_genius_song](#), 4
[get_genius_song_lyrics](#), 5
[get_song_media](#), 6
[get_song_producer_artist](#), 7
[get_song_relationship](#), 8

[search_genius_artist](#), 8
[search_genius_song](#), 9