

Package ‘progRchives’

October 27, 2021

Title Scrape And Organize ProgArchives Data

Version 0.0.2

Description A collection of functions for scraping and organizing data from ProgArchives (<http://www.progarchives.com/>).

Depends R (>= 4.0.0)

License GPL-2

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Imports rvest, purrr, dplyr, stringr, tidyr, tidyselect, janitor, lubridate, future, furrr

R topics documented:

extract_album_data	1
extract_artist_urls	2
extract_ratings	2
extract_subgenres	3
progarchives	3
scrape_progarchives	4
Index	5

extract_album_data	<i>Extract Artist and Album Info from Artist Page</i>
--------------------	-------------------------------------------------------

Description

Scrape artist name and album data from an artist page as a data frame

Usage

```
extract_album_data(url_artist)
```

Arguments

url_artist An artist URL, preferably those extracted with [extract_artist_urls](#).

Examples

```
## Not run: extract_album_data(url_artist = "http://www.progarchives.com/artist.asp?id=4398")
## Not run: extract_album_data(url_artist = "http://www.progarchives.com/artist.asp?id=671")
```

extract_artist_urls	<i>Extract Artist URL and Country from Subgenre Page</i>
---------------------	----------------------------------------------------------

Description

Scrape all artist links and countries from a subgenre page as a dataframe

Usage

```
extract_artist_urls(url_genre)
```

Arguments

url_genre A subgenre URL, preferably those extracted with [extract_subgenres](#).

Examples

```
## Not run: extract_artist_urls(url_genre = "http://www.progarchives.com/subgenre.asp?style=12")
## Not run: extract_artist_urls(url_genre = "http://www.progarchives.com/subgenre.asp?style=35")
```

extract_ratings	<i>Extract Each Album Rating</i>
-----------------	----------------------------------

Description

Scrape number of ratings and weight of each rating as a dataframe (simple ratings have weight 1, reviews have weight 10 and collaborator reviews have weight 20).

Usage

```
extract_ratings(url_album)
```

Arguments

url_album An album URL, preferably those extracted with [extract_album_data](#).

Examples

```
extract_ratings(url_album = "http://www.progarchives.com/album.asp?id=69658")
extract_ratings(url_album = "http://www.progarchives.com/album.asp?id=22272")
```

extract_subgenres	<i>Extract All Subgenre URLs</i>
-------------------	----------------------------------

Description

Scrape all subgenre names and URLs from **ProgArchives** as a dataframe

Usage

```
extract_subgenres()
```

Examples

```
## Not run: extract_subgenres()
```

progarchives	<i>ProgArchives Data from 2021-10-18</i>
--------------	------------------------------------------

Description

A dataset containing album data by nearly 12,000 Progressive Rock artists.

Usage

```
progarchives
```

Format

A data frame with 66507 rows and 12 variables:

date Date of extraction (2021-10-18)

subgenre Subgenre name

url_subgenre Subgenre URL

artist Band/artist name; different artists might share a name

country Country from which the artist hails

url_artist Artist URL

album Album name; different release might share a name; some artists have no registered release

url_album Album URL

type Type of release: Studio Album, Live, DVD/Video, Boxset/Compilation, Singles/EPs/Fan Club/Promo or NA

year Release year

avg_rating Average rating, ranging from 1.00 to 5.00, including NAs

n_ratings Number of ratings

Source

<http://www.progarchives.com/>

scrape_progarchives	<i>Scrape All Album Data from ProgArchives</i>
---------------------	------------------------------------------------

Description

Extract subgenre, artist and album data from **ProgArchives**

Usage

```
scrape_progarchives(parallel = TRUE)
```

Arguments

parallel	Logical value; should the function use parallelism when extracting album data? Defaults to TRUE.
----------	-----------------------------------------------------------------------------------------------------

Examples

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

Index

* **datasets**

progarchives, [3](#)

extract_album_data, [1](#), [2](#)

extract_artist_urls, [1](#), [2](#)

extract_ratings, [2](#)

extract_subgenres, [2](#), [3](#)

progarchives, [3](#)

scrape_progarchives, [4](#)