Package ‘appler’

June 13, 2023

Type  Package

Title  'Apple App Store' and 'iTunes' Data Extraction

Version  0.2.1


URL  https://ashbaldry.github.io/appler/,

https://performance-partners.apple.com/search-api

BugReports  https://github.com/ashbaldry/appler/issues

Depends  R (>= 2.10)

Imports  httr, jsonlite, rvest

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

License  GPL-2

Encoding  UTF-8

RoxygenNote  7.2.3

VignetteBuilder  knitr

LazyData  true

Language  en-GB

NeedsCompilation  no

Author  Ashley Baldry [aut, cre]

Maintainer  Ashley Baldry <arbaldry91@gmail.com>

Repository  CRAN

Date/Publication  2023-06-13 07:00:08 UTC
**apple_apps**

### Description

A dataset containing a selection of apps available on the Apple App Store with a corresponding ID that can be used in `appler` functions.

To see more information about the application online, you can add the following URL in your browser: `apps.apple.com/app/id<id>` where `<id>` is the 'app_id' column.

### Usage

```r
apple_apps
```

### Format

A data frame with 2 columns and 202 rows

- **app_name** Application name
- **app_id** Apple ID of the application

### Details

All of the applications in this table are available in Canada (`country_id = "ca"`) at the time of writing (2022-12-03), however they might not be available in all countries, or have a different application name.

### Source

<https://apps.apple.com>
Examples

# Get information about Microsoft Teams
teams <- apple_apps[apple_apps$app_name == "Microsoft Teams", "app_id"]

# Search for any other apps
search_apple(term = "Microsoft Teams", country = "ca", media = "software")

# General application information including average rating
lookup_apple(teams, country = "ca")

# Latest application reviews
get_apple_reviews(teams, country = "ca")

# Current position on App store
get_apple_chart_position(teams, country = "ca")

---

get_apple_chart_position

Apple App Store Chart Position

Description

Search for whether an application is currently in the top 100 apps of any category on the Apple App Store.

Usage

get_apple_chart_position(id, country)

Arguments

id

The ID of the App on the Apple App Store. Either found by using search_apple, or available in the URL of the app to pull reviews from. For example, GitHub's App ID is 1477376903, as seen in its URL: https://apps.apple.com/gb/app/id1477376905

country

The two-letter country code for the store you want to search. For a list of country codes see https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2

Value

A list of two, containing the ‘position’ and the ‘category’ of the app if available.

If the application is not in the charts then both fields will return as NA
get_apple_rating_split

Apple App Store Ratings

Description
Scrapes the App store page and retrieves the split of the ratings between 1 and 5 stars

Usage
get_apple_rating_split(id, country = "us")

Arguments
id The ID of the App on the Apple App Store
country The two-letter country code for the store you want to search. For a list of country

Details
For overall rating and count, use lookup_apple

Value
A 5 row data.frame with the split of 1-5 stars given rounded to the nearest percent

Examples

# Search for GitHub in App Store in the UK
country_id <- "gb"
github_search_results <- search_apple(
  term = "GitHub",
  country = country_id,
  media = "software"
)

# Look up chart position for GitHub in the UK
# (App ID found in trackId column of github_search_results)
get_apple_chart_postion(1477376905, "gb")
get_apple_reviews

country = country_id,
media = "software"
)

# Look up app store rating split for GitHub in the UK
# (App ID found in trackId column of github_search_results)
get_apple_rating_split(1477376905, country_id)

---

get_apple_reviews Apple App Store Reviews

Description

Using Apple’s RSS feed, extract the most recent or helpful reviews for a specific application.

Usage

get_apple_reviews(
  id,
  country = "us",
  all_results = FALSE,
  page_no = 1,
  sort_by = c("mostrecent", "mosthelpful")
)

Arguments

id The ID of the App on the Apple App Store. Either found by using search_apple, or available in the URL of the app to pull reviews from. For example, GitHub’s App ID is 1477376903, as seen in its URL: https://apps.apple.com/gb/app/id1477376905

country The two-letter country code for the store you want to search. For a list of country codes see https://en.wikipedia.org/wiki/ISO_3166-1_alpha-2

all_results Logical, would you like all possible reviews to be pulled? By default set to FALSE

page_no If page_no = FALSE then the page of reviews to pull. Defaults to most recent.

sort_by Which order should the reviews be pulled? There are currently two possible options:

"mostrecent" Sorts by the time reviews are posted and pulls the most recently posted reviews

"mosthelpful" Sorts the reviews by usefulness and returns the most useful posts. For larger apps, the top 500 may not match the top 500 most recent
Details

There is a limitation in Apple’s RSS feed that means only the 500 most recent/helpful reviews can be pulled. There are 10 pages of results from the RSS feed, each one containing 50 reviews. It is recommended to periodically store reviews in a database or other storage system to track the older reviews.

Value

A data.frame of the extracted reviews, containing:

- id: The review ID
- review_time: The time the review was posted on the App Store
- author: The username of the reviewer
- app_version: The version of the application that was installed when reviewing the application
- title: Title summary of the review
- rating: The rating (out of 5) given to the application
- review: The text of the review

If there were no reviews then it will return NULL.

Examples

```r
# Search for GitHub in App Store in the UK
country_id <- "gb"
github_search_results <- search_apple(  
  term = "GitHub",  
  country = country_id,  
  media = "software"
)

# Look up reviews for GitHub  
# (App ID found in trackId column of github_search_results)  
get_apple_reviews(1477376905, country_id)
```

__itunes_artists__

**iTunes Artists**

**Description**

A small dataset containing current artists available on iTunes with a corresponding ID that can be used in appler functions.

To see more information about the artist online, you can add the following URL in your browser:
music.apple.com/artist/<id> where <id> is the ‘artist_id’ column
lookup_apple

Usage

itunes_artists

Format

A data frame with 2 columns and 10 rows

artist  Artist name
artist_id  Apple ID of the artist

Source

<https://music.apple.com>

Examples

# Get information about Microsoft Teams
lizzo <- itunes_artists[itunes_artists$artist == "Lizzo", ]
lizzo_id <- lizzo$artist_id
lizzo_name <- lizzo$artist

# Search for artist by name, can find the ID from this query
search_apple(term = lizzo_name, country = "ca", lang = "en")

# Get information about the artist
lookup_apple(id = lizzo_id, country = "ca", sort = "recent")

Description

You can create a lookup request to search for content in the stores based on iTunes IDs, UPCs/EANs, and All Music Guide (AMG) IDs. ID-based lookups are faster and contain fewer false-positive results.

Usage

lookup_apple(
    id,
    country = NULL,
    entity = NULL,
    limit = NULL,
    sort = NULL,
    id_type = "id"
  )
Arguments

- **id**: The ID of the iTunes entity
- **entity**: Optional The type of results you want returned, relative to the specified media type.
- **limit**: Optional The number of search results you want the iTunes Store to return between 1 and 200. The default is 50.
- **sort**: Optional The order the results are returned, for most recent first select recent.
- **id_type**: The ID type to lookup, options are:
  - **id**: The default iTunes ID
  - **amgArtistId**: AMG Artist ID
  - **amgAlbumId**: AMG Album ID
  - **upc**: UPC Album or Video ID
  - **isbn**: ISB Book ID

Value

A `data.frame` of any results that match the iTunes database.

If there were no successful results then it will return `NULL`.

See Also


Examples

```r
# Search for all Jack Johnson audio and video content
search_apple(term = "Jack Johnson")

# Look up Jack Johnson by iTunes artist ID
lookup_apple(909253)

# Look up Jack Johnson by AMG artist ID
lookup_apple(468749, id_type = "amgArtistId")
```
search_apple  Apple Store Search

Description

Using Apple’s iTunes API, will find any content available from Apple based on a given search term.

Usage

```r
search_apple(
  term,
  country = NULL,
  media = NULL,
  entity = NULL,
  attribute = NULL,
  limit = NULL,
  lang = c("en_us", "ja_jp"),
  explicit = c("Yes", "No")
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>term</td>
<td>The URL-encoded text string you want to search for. For example: <code>jack+johnson</code>. If using a vector it will search for all terms in one search.</td>
</tr>
<tr>
<td>media</td>
<td>Optional The media type you want to search for. For example: movie. The default is all.</td>
</tr>
<tr>
<td>entity</td>
<td>Optional The type of results you want returned, relative to the specified media type.</td>
</tr>
<tr>
<td>attribute</td>
<td>Optional The attribute you want to search for in the stores, relative to the specified media type. For example, if you want to search for an artist by name specify <code>entity=allArtist&amp;attribute=allArtistTerm</code>. In this example, if you search for <code>term=maroon</code>, iTunes returns &quot;Maroon 5&quot; in the search results, instead of all artists who have ever recorded a song with the word &quot;maroon&quot; in the title.</td>
</tr>
<tr>
<td>limit</td>
<td>Optional The number of search results you want the iTunes Store to return between 1 and 200. The default is 50.</td>
</tr>
<tr>
<td>lang</td>
<td>Optional The language, English or Japanese, you want to use when returning search results.</td>
</tr>
<tr>
<td>explicit</td>
<td>Optional A flag indicating whether or not you want to include explicit content in your search results.</td>
</tr>
</tbody>
</table>
Value

A `data.frame` of any results that match the iTunes database. If there were no successful results then it will return `NULL`.

See Also

https://performance-partners.apple.com/search-api

Examples

```r
# Search for all Jack Johnson audio and video content
search_apple(term = "jack johnson")

# To search for all Jack Johnson audio and video content and return only the first 25 items
search_apple(term = "jack johnson", limit = 25)
```
Index

* datasets
  apple_apps, 2
  itunes_artists, 6

apple_apps, 2

get_apple_chart_position, 3
get_apple_rating_split, 4
get_apple_reviews, 5

itunes_artists, 6

lookup_apple, 4, 7

search_apple, 3, 5, 9