spotify-manager Documentation

Release 1.3

Marc Solé

Feb 21, 2019

Contents

spotify-manager is a lightweight Python3 library for the easy use and integration of Spotipy in your projects. As long as this library is only an upper leveled Spotipy, it still works with the Spotify Web API and your are going to need Spotify Premium and an API Token.

The spirit of spotify-manager is to automatize some requests related with the streaming, but not all of them. Searching data or updating your profile is less important for this library than playing your fifty last tracks again or repeat a song. I just focused on the things I find more important and useful.

Here's a quick example of using *spotify-manager* to search and play a track like for example 'Mockingbird' by 'Eminem':

```
from spotify_manager import SpotifyManager

sm = SpotifyManager(username, client_id, client_secret, redirect_uri)
sm.play_song('eminem mockingbird')
```

Here's another example showing how to play songs related to the artist you are listening to right now:

```
from spotify_manager import SpotifyManager

sm = SpotifyManager(username, client_id, client_secret, redirect_uri)
sm.play_similar_from_current_artist()
```

Finally, here's an example of increasing the volume of the device you are listening to right now a twenty percent:

```
from spotify_manager import SpotifyManager

sm = SpotifyManager(username, client_id, client_secret, redirect_uri)
sm.add_volume(20)
```

As you can see, it's pretty simple to use this library and it's more upper leveled than Spotipy, so you don't have to focus that much on the implementation.

CHAPTER 1

Features

spotify-manager is a lightweight Python3 library for the easy use and integration of Spotipy in your projects. As long as this library is only an upper leveled Spotipy, it still works with the Spotify Web API and your are going to need Spotify Premium and an API Token.

The spirit of spotify-manager is to automatize some requests related with the streaming, but not all of them. Searching data or updating your profile is less important for this library than play your fifty last tracks again or repeat a song. I just focused on the things I find more important and util.

CHAPTER 2

Installation

Install *spotify-manager* with:

```
pip3 install spotify-manager
```

Or with:

```
easy_install spotify-manager
```

Or you can get the source from github at https://github.com/WolfyLPDC/spotify-manager

CHAPTER 3

Authorized requests

As long as this is a library that uses Spotify's API you are going to need authentication to use it. Because it has the same problems than Spotipy, and I'm using that library, the best I can do is to link their docs to allow you to read about this problem (feature).

https://spotipy.readthedocs.io/en/latest/#authorized-requests

CHAPTER 4

API Reference

CHAPTER 5

spotify_manager Module

class spotify_manager.spotify_manager.**SpotifyManager**(*username*, *client_id*, *client_secret*, *redirect_uri*)

Bases: object

__init__(*username*, *client_id*, *client_secret*, *redirect_uri*)

Create a SpotifyManager object.

Parameters

- **username** – The Spotify Premium username.
- **client_id** – The client id of your app.
- **client_secret** – The client secret of your app.
- **redirect_uri** – The redirect URI of your app.

decrease_volume(*volume_percent*, *device_id=None*)

Decreases device's volume in percentage.

Parameters

- **volume_percent** – Volume percentage to decrease. Negative to increase.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – volume_percent is not an integer.

delete_current_album()

Saves current album on user's library.

Raises **ConnectionError** – User is not connected to Spotify

delete_current_song()

Deletes current song from user's library.

Raises **ConnectionError** – User is not connected to Spotify

get_current_album_info()
Gets information about current song's album.

Returns Dictionary.

Raises **ConnectionError** – User is not connected to Spotify.

get_current_album_release_date()
Gets release year from current song's album.

Returns Release date as a string with format YYYY-MM-DD.

Raises **ConnectionError** – User is not connected to Spotify.

get_current_song_artist()
Gets artists from current song.

Returns String of artists names separated by commas.

Raises **ConnectionError** – User is not connected to Spotify.

get_current_song_info()
Gets information about current song.

Returns Dictionary.

Raises **ConnectionError** – User is not connected to Spotify.

get_repeat_state()
Gets repeat state.

Returns Repeat state, which can be 'track', 'context' or 'off'.

Raises **ConnectionError** – User is not connected to Spotify.

get_shuffle_state()
Gets shuffle state.

Returns Repeat state, which can be True or False.

Raises **ConnectionError** – User is not connected to Spotify.

get_volume(*device_id=None*)
Returns device's volume in percentage.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

increase_volume(*volume_percent*, *device_id=None*)
Increases device's volume in percentage.

Parameters

- **volume_percent** – Volume percentage to increase. Negative to decrease.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – volume_percent is not an integer.

next_repeat_state(*device_id=None*)
Moves repeat state to next state.

Order is 'track' -> 'context' -> 'off' -> 'track'.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – User is not connected to Spotify.

next_song(*device_id=None*)
Moves playback to next song.

Doesn't throw an error if there is no active device.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

pause(*device_id=None*)
Pauses device's playback.

Doesn't throw an error if there is no active device.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

play(*device_id=None*)
Starts or resumes device's playback.

Doesn't throw an error if there is no active device.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

play_album(*album_name*, *device_id=None*)
Search album that matches album_name and plays it.

Doesn't throw an error if there is no active device.

Parameters

- **album_name** – Query to match.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – There is no search query.
- **IndexError** – There is no results.

play_artist(*artist_name*, *device_id=None*)
Search artist that matches song_name and plays it.

Doesn't throw an error if there is no active device.

Parameters

- **artist_name** – Query to match.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – There is no search query.
- **IndexError** – There is no results.

play_genre(*genre_name*, *limit=20*, *device_id=None*)
Search genre that matches genre_name and plays it.

Doesn't throw an error if there is no active device.

Parameters

- **genre_name** – Query to match.
- **limit** – Number of songs to search and play.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – genre_name is not valid. Also limit is not an integer.

play_playlist(*playlist_name*, *device_id=None*)
Search playlist that matches playlist_name and plays it.

Doesn't throw an error if there is no active device.

Parameters

- **playlist_name** – Query to match.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – device_id is not valid.
- **TypeError** – There is no search query.
- **IndexError** – There is no results.

play_recently_played(*limit=50*, *device_id=None*)
Search songs that user played recently.

Parameters

- **limit** – Number of songs to search and play.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid. Also User is not connected to Spotify.
- **TypeError** – limit is not an integer.

play_similar_from_current_artist(*limit=20*, *device_id=None*)
Search songs from similar artists of the current one and play them.

Parameters

- **limit** – Number of songs to search and play.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid. Also User is not connected to Spotify.
- **TypeError** – limit is not an integer.

play_similar_from_current_track(*limit=20*, *device_id=None*)
Search songs similar to the current one and play them.

Parameters

- **limit** – Number of songs to search and play.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid. Also User is not connected to Spotify.
- **TypeError** – limit is not an integer.

play_song(*song_name*, *device_id=None*)
Search song that matches song_name and plays it.

Doesn't throw an error if there is no active device.

Parameters

- **song_name** – Query to match.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – device_id is not valid.
- **TypeError** – There is no search query.
- **IndexError** – There is no results.

play_top_artists(*limit=5*, *device_id=None*)
Search top songs from artists that user plays the most and plays them.

Parameters

- **limit** – Number of artists to analyze.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid. Also User is not connected to Spotify.
- **TypeError** – limit or offset are not an integer.

play_top_tracks(*limit=20*, *device_id=None*)
Search songs that user plays the most and plays them.

Parameters

- **limit** – Number of songs to search and play.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid. Also User is not connected to Spotify.
- **TypeError** – limit or offset are not an integer.

previous_song(*restart_time=0*, *device_id=None*)
Moves playback to previous song. If there is no previous the actual one is restarted.

If song's peek is greater than restart_time, song is moved instead of restarted.

Doesn't throw an error if there is no active device.

Parameters

- **restart_time** – Minimum time in seconds to restart song instead of move playback. 0 to disable.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – volume_percent is not an integer.

restart_song(*device_id=None*)
Restarts current song.

Doesn't throw an error if there is no active device.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

save_current_album()
Saves current album on user's library.

Raises **ConnectionError** – User is not connected to Spotify

save_current_song()
Saves current song on user's library.

Raises **ConnectionError** – User is not connected to Spotify

set_repeat_state(*repeat_state*, *device_id=None*)
Sets repeat state.

Doesn't throw an error if there is no active device.

Parameters

- **repeat_state** – Repeat state, which can be 'track', 'context' or 'off'.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – User is not connected to Spotify.
- **TypeError** – Repeat state must be 'track', 'context' or 'off'.

set_shuffle_state(*shuffle_state*, *device_id=None*)
Sets shuffle state.

Doesn't throw an error if there is no active device.

Parameters

- **shuffle_state** – Shuffle state, which can be True or False.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – User is not connected to Spotify.
- **TypeError** – Shuffle state must be True or False.

set_volume(*volume_percent*, *device_id=None*)
Sets device's volume to new percentage.

Doesn't throw an error if there is no active device.

Parameters

- **volume_percent** – Volume percentage to set.
- **device_id** – Device target, if it's not set, target is current device.

Raises

- **ConnectionError** – There is no active device or device_id is not valid.
- **TypeError** – volume_percent is not an integer.

switch_play_pause(*device_id=None*)
Switch between Play and Pause state.

Doesn't throw an error if there is no active device.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – There is no active device or device_id is not valid.

switch_shuffle_state(*device_id=None*)
Switch shuffle state between True and False.

Parameters **device_id** – Device target, if it's not set, target is current device.

Raises **ConnectionError** – User is not connected to Spotify.

CHAPTER 6

Support

If you any have questions about spotify-manager, you can mail me to my account *'marcsole @ insomniacwolves.com'* and I will try to answer as soon as possible.

If you think you've found a bug, let me know at spotify-manager

CHAPTER 7

Contribute

spotify-manager authored by Marc Solé (WolfyLPDC) with special thanks to Paul Lamere (plamere), the author of the great library Spotipy, and to all the people that have contributed to make this possible.

CHAPTER 8

License

https://github.com/WolfyLPDC/spotify-manager/blob/master/LICENSE

Python Module Index

S

Index

Symbols

D

G

I

N

P

R

S