Infnet

Projeto de Bloco: Ciência de Dados Aplicada [24E3_5]

Aluno: Rodrigo Avila

Para acessar o projeto no GIT Hub, clique aqui

Web Scrapping Genius.com

A ideia é fazer o webscrapping de letras da banda Metallica no site genius.com

Após analisar o site, a páginação é feita através de um scrolling infinito, ou seja, a cada vez que o usuário chega no final da página, mais músicas são carregadas.

O que dificulta fazer o scrapping de todas as músicas, mas ao analisar a parte de network da página, percebi que a página faz requisições para uma API no backend utilizando paginação, por mais que o link não apareça na url.

Exemplo de requisição para a API:

```
https://genius.com/api/artists/10662/songs?
page=2&per_page=20&sort=popularity&text_format=html%2Cmarkdown
```

Dessa forma, descobri que a banda Metallica possui o ID 10662, e que a cada requisição, 20 músicas são retornadas, não sendo necessário realizar o scrapping. É possível chamar o backend diretamente.

Porém, para atender o requisito do exercício, será feito um scrapping do album Master of Puppets, que possui 8 músicas, esse album é um bom exemplo, pois possui uma música instrumental, o que exige um tratamento diferenciado.

In [20]:
```python
import requests
from bs4 import BeautifulSoup

class GeniusAlbumSongsCrawler:
    def __init__(self, album_url: str):
        self.album_url = album_url
        self._soup = None

    @property
    def soup(self) -> BeautifulSoup:
        return self._soup
```

```
    def fetch_page(self) -> BeautifulSoup:
        response = requests.get(self.album_url)
        response.raise_for_status()
        self._soup = BeautifulSoup(response.text, 'html.parser')
        return self._soup

    def extract_song_links(self) -> list:
        if not self._soup:
            raise ValueError("Soup not initialized. Call fetch_page() first.

        song_links = []
        for link in self._soup.select('a.u-display_block'):
            href = link.get('href')
            if href:
                song_links.append(href)
        return song_links

crawler = GeniusAlbumSongsCrawler('https://genius.com/albums/Metallica/Maste
crawler.fetch_page()
links = crawler.extract_song_links()
links
```

Out[20]:
```
['https://genius.com/Metallica-battery-lyrics',
 'https://genius.com/Metallica-master-of-puppets-lyrics',
 'https://genius.com/Metallica-the-thing-that-should-not-be-lyrics',
 'https://genius.com/Metallica-welcome-home-sanitarium-lyrics',
 'https://genius.com/Metallica-disposable-heroes-lyrics',
 'https://genius.com/Metallica-leper-messiah-lyrics',
 'https://genius.com/Metallica-orion-lyrics',
 'https://genius.com/Metallica-damage-inc-lyrics']
```

In [21]:
```
import requests
from bs4 import BeautifulSoup

class GeniusSongLyricsScraper:
    def __init__(self, song_url: str):
        self.song_url = song_url
        self._soup = None

    @property
    def soup(self) -> BeautifulSoup:
        if not self._soup:
            self.fetch_page()
        return self._soup

    def fetch_page(self) -> None:
        response = requests.get(self.song_url)
        response.raise_for_status()
        self._soup = BeautifulSoup(response.text, 'html.parser')

    def extract_lyrics(self) -> str:
        instrumental_message = self.soup.select_one('div.LyricsPlaceholder__
        if instrumental_message and 'This song is an instrumental' in instru
            return 'This song is an instrumental'
```

```
        lyrics_div = self.soup.select_one('div[class*="Lyrics__Container"]')
        if not lyrics_div:
            raise ValueError("Lyrics div not found.")

        lyrics = []
        for element in lyrics_div.descendants:
            if element.name == 'br':
                lyrics.append('\n')
            elif isinstance(element, str):
                lyrics.append(element)

        return ''.join(lyrics)

    def extract_song_name(self) -> str:
        song_name_span = self.soup.select_one('h1 span[class*="SongHeaderdes
        if not song_name_span:
            raise ValueError("Song name span not found.")

        return song_name_span.text

    def extract_album_name(self) -> str:
        album_name_div = self.soup.select_one('div.HeaderArtistAndTracklistc
        if not album_name_div:
            raise ValueError("Album name div not found.")

        return album_name_div.text.strip()

    def extract_artist_name(self) -> str:
        artist_name_div = self.soup.select_one('div.HeaderArtistAndTracklist
        if not artist_name_div:
            raise ValueError("Artist name div not found.")

        return artist_name_div.text.strip()

# Example usage:
scraper = GeniusSongLyricsScraper("https://genius.com/Metallica-battery-lyri
lyrics = scraper.extract_lyrics()
song_name = scraper.extract_song_name()
album_name = scraper.extract_album_name()
artist_name = scraper.extract_artist_name()

print(f"Album Name: {album_name}")
print(f"Song Name: {song_name}")
print(f"Artist Name: {artist_name}\n")
print(f"Lyrics:\n\n{lyrics}")
```

```
Album Name: Master of Puppets (Deluxe Box Set)
Song Name: Battery
Artist Name: Metallica

Lyrics:

[Verse 1]
Lashing out the action, returning the reaction
Weak are ripped and torn away
Hypnotizing power, crushing all that cower
Battery is here to stay

[Chorus]
Smashing through the boundaries, lunacy has found me
Cannot stop the battery
Pounding out aggression, turns into obsession
Cannot kill the battery
Cannot kill the family, battery is found in me
Battery, battery

[Verse 2]
Crushing all deceivers, mashing non-believers
Never-ending potency
Hungry violence-seeker, feeding off the weaker
Breeding on insanity

[Chorus]
Smashing through the boundaries, lunacy has found me
Cannot stop the battery
Pounding out aggression, turns into obsession
Cannot kill the battery
Cannot kill the family, battery is found in me
Battery, battery
```

In [3]:

```python
import json
import csv
from typing import Literal

def save_genius_album_lyrics_to_file(
    file_path: str,
    album_url: str,
    format: Literal['json', 'csv'] = 'csv'):

    crawler = GeniusAlbumSongsCrawler(album_url)
    crawler.fetch_page()
    song_links = crawler.extract_song_links()

    songs_list = []

    for song_url in song_links:
        print(f"Processing {song_url}")
        scraper = GeniusSongLyricsScraper(song_url)
        scraper.fetch_page()
        song_name = scraper.extract_song_name()
        album_name = scraper.extract_album_name()
```

```
        artist_name = scraper.extract_artist_name()
        lyrics = scraper.extract_lyrics()

        song_info = {
            'album_name': album_name,
            'song_name': song_name,
            'artist_name': artist_name,
            'lyrics': lyrics
        }
        songs_list.append(song_info)

    if format == 'json':
        with open(file_path, 'w', encoding='utf-8') as json_file:
            json.dump(songs_list, json_file, ensure_ascii=False, indent=4)
    elif format == 'csv':
        with open(file_path, 'w', encoding='utf-8', newline='') as csv_file:
            writer = csv.DictWriter(csv_file, fieldnames=['album_name', 'sor
            writer.writeheader()
            for song_info in songs_list:
                writer.writerow(song_info)

    print(f"Lyrics saved to {file_path}")

save_genius_album_lyrics_to_file(
    '../data/processed/metallica_master_of_puppets_lyrics.csv',
    'https://genius.com/albums/Metallica/Master-of-puppets',
    'csv'
)
```

```
Processing https://genius.com/Metallica-battery-lyrics
Processing https://genius.com/Metallica-master-of-puppets-lyrics
Processing https://genius.com/Metallica-the-thing-that-should-not-be-lyrics
Processing https://genius.com/Metallica-welcome-home-sanitarium-lyrics
Processing https://genius.com/Metallica-disposable-heroes-lyrics
Processing https://genius.com/Metallica-leper-messiah-lyrics
Processing https://genius.com/Metallica-orion-lyrics
Processing https://genius.com/Metallica-damage-inc-lyrics
Lyrics saved to ../data/processed/metallica_master_of_puppets_lyrics.csv
```

In [4]:

```
import pandas as pd

df = pd.read_csv('../data/processed/metallica_master_of_puppets_lyrics.csv')

df.head()
```

Out[4]:

	album_name	song_name	artist_name	lyrics
0	Master of Puppets (Deluxe Box Set)	Battery	Metallica	[Verse 1]\nLashing out the action, returning t...
1	Master of Puppets (Deluxe Box Set)	Master of Puppets	Metallica	[Verse 1]\nEnd of passion play, crumbling away...
2	Master of Puppets (Deluxe Box Set)	The Thing That Should Not Be	Metallica	[Verse 1]\nMessenger of fear in sight\nDark de...
3	Master of Puppets (Deluxe Box Set)	Welcome Home (Sanitarium)	Metallica	[Verse 1]\nWelcome to where time stands still\...
4	Master of Puppets (Deluxe Box Set)	Disposable Heroes	Metallica	[Verse 1]\nBodies fill the fields I see, hungr...

In [5]:
```
save_genius_album_lyrics_to_file(
    '../data/processed/metallica_and_justice_for_all_lyrics.csv',
    'https://genius.com/albums/Metallica/And-justice-for-all',
    'csv'
)
```

```
Processing https://genius.com/Metallica-blackened-lyrics
Processing https://genius.com/Metallica-and-justice-for-all-lyrics
Processing https://genius.com/Metallica-eye-of-the-beholder-lyrics
Processing https://genius.com/Metallica-one-lyrics
Processing https://genius.com/Metallica-the-shortest-straw-lyrics
Processing https://genius.com/Metallica-harvester-of-sorrow-lyrics
Processing https://genius.com/Metallica-the-frayed-ends-of-sanity-lyrics
Processing https://genius.com/Metallica-to-live-is-to-die-lyrics
Processing https://genius.com/Metallica-dyers-eve-lyrics
Lyrics saved to ../data/processed/metallica_and_justice_for_all_lyrics.csv
```

In [22]:
```
save_genius_album_lyrics_to_file(
    '../data/processed/guns_n_roses_ride_greatest_hits_lyrics.csv',
    'https://genius.com/albums/Guns-n-roses/Greatest-hits',
    'csv'
)
```

```
Processing https://genius.com/Guns-n-roses-welcome-to-the-jungle-lyrics
Processing https://genius.com/Guns-n-roses-sweet-child-o-mine-lyrics
Processing https://genius.com/Guns-n-roses-patience-lyrics
Processing https://genius.com/Guns-n-roses-paradise-city-lyrics
Processing https://genius.com/Guns-n-roses-knockin-on-heavens-door-lyrics
Processing https://genius.com/Guns-n-roses-civil-war-lyrics
Processing https://genius.com/Guns-n-roses-you-could-be-mine-lyrics
Processing https://genius.com/Guns-n-roses-dont-cry-lyrics
Processing https://genius.com/Guns-n-roses-november-rain-lyrics
Processing https://genius.com/Guns-n-roses-live-and-let-die-lyrics
Processing https://genius.com/Guns-n-roses-yesterdays-lyrics
Processing https://genius.com/Guns-n-roses-aint-it-fun-lyrics
Processing https://genius.com/Guns-n-roses-since-i-dont-have-you-lyrics
Processing https://genius.com/Guns-n-roses-sympathy-for-the-devil-lyrics
Lyrics saved to ../data/processed/guns_n_roses_ride_greatest_hits_lyrics.csv
```

In [ ]:
```
!playwright install
```

In [3]:
```
#Exportando para PDF
!jupyter nbconvert --to webpdf rodrigo_avila_PB_TP2.ipynb
```

```
[NbConvertApp] Converting notebook rodrigo_avila_PB_TP2.ipynb to webpdf
[NbConvertApp] Building PDF
[NbConvertApp] PDF successfully created
[NbConvertApp] Writing 240897 bytes to rodrigo_avila_PB_TP2.pdf
```