УДК 004

Создание парсера на языке Python с использованием библиотеки BeautifulSoup

Болтовский Гавриил Александрович
Приамурский государственный университет им. Шолом-Алейхема
Студент

Аннотация
Целью данной статьи является создание парсера на языке Python с использованием библиотеки BeautifulSoup. Собираются тексты песен с сайта Genius. Результатом исследования станет готовая программа с подробным описанием её создания.
Ключевые слова: парсинг, BeautifulSoup, Python

Creating a Python parser using the BeautifulSoup library

Boltovskiy Gavriil Aleksandrovich
Sholom-Aleichem Priamursky State University
Student

Abstract
The purpose of this article is to create a Python parser using the BeautifulSoup libraries. Collects lyrics from the Genius website. The result of the study will be a finished program with a detailed description of its creation.
Keywords: parsing, BeautifulSoup, Python

1. Введение

1.1 Актуальность исследования

Парсинг данных подразумевает сбор каких-либо данных с целью их последующей обработки. Язык программирования Python позволяет выполнять эту задачу, для этого существуют специальные библиотеки, такие как requests, которая позволяет работать с HTTP, BeautifulSoup, которая даёт функционал в исследовании HTML кода страницы. Таким образом можно получать разнообразную информацию со страниц. В рамках данной статьи будет рассмотрен сбор текста песен с сайта Genius [1].

1.2 Обзор исследований

Задачи парсинга данных имеет свои реализации и на других языках программирования. Обзор библиотек на C# проведён С.В. Ивановой [2]. В C# используются такие библиотеки как AngleSharp, HtmlAgilityPack, CSQuery и другие. В исследовании И.И. Карабак, К.А. Зорина, [3] рассматриваются примеры парсинга в публичных сообществах мессенджера «Телеграм». В противовес парсингу решаются проблемы защиты информации от утечки,

защиты серверов компаний от нагрузки, которые оказывают парсинг-программы. Последствия копирования данных из интернет ресурсов, описание методов защиты описываются в статье А.И. Дубровиной [4]. Х.И. Эшонкулов в своём исследовании [5] так же изучает проблему необходимости защиты от автоматизированного сбора информации с веб-ресурсов. Вопрос потребности защиты от парсинга получил всестороннее рассмотрение в исследовании А.А. Менщикова, А.В. Комарова, Ю.А. Гатчина [6].

1.3 Цель исследования

Целью исследование является создание программы для автоматизированного сбора текстов песен с интернет ресурса Genius.

1.4 Постановка задачи

Реализована программа будет на языке программирования Python. Для загрузки HTML кода страницы будет использоваться библиотека requests, а для его обработки beautifulsoup4. Разработка будет происходить в IDE PyCharm

2. Методы исследования

Перед началом работы следует установить библиотеки (рис. 1). Это можно сделать из самой IDE.

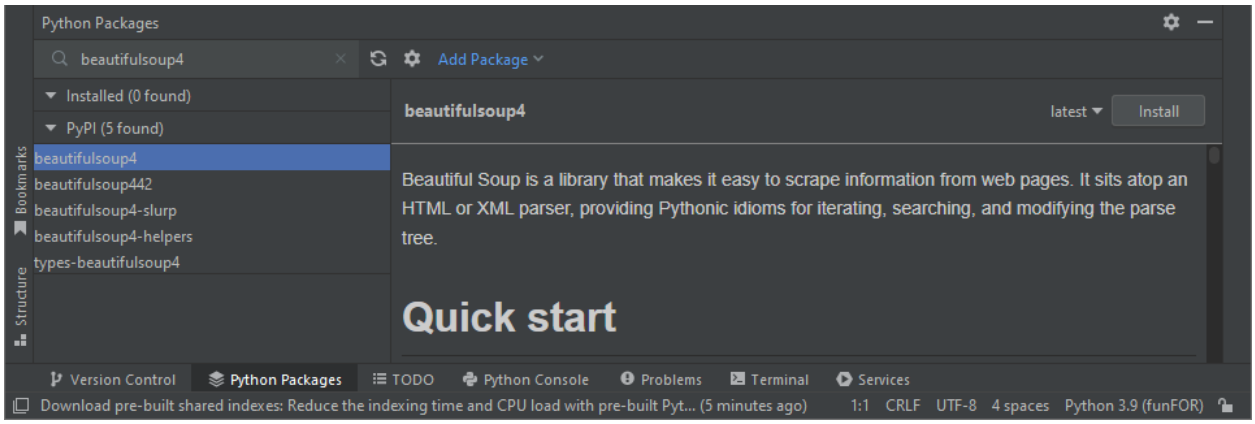

Рисунок 1 – Установка beautifulsoup4

Аналогично устанавливается библиотека requests.

Requests необходима для загрузки страницы. Для начала необходимо сделать get-запрос (рис. 2).

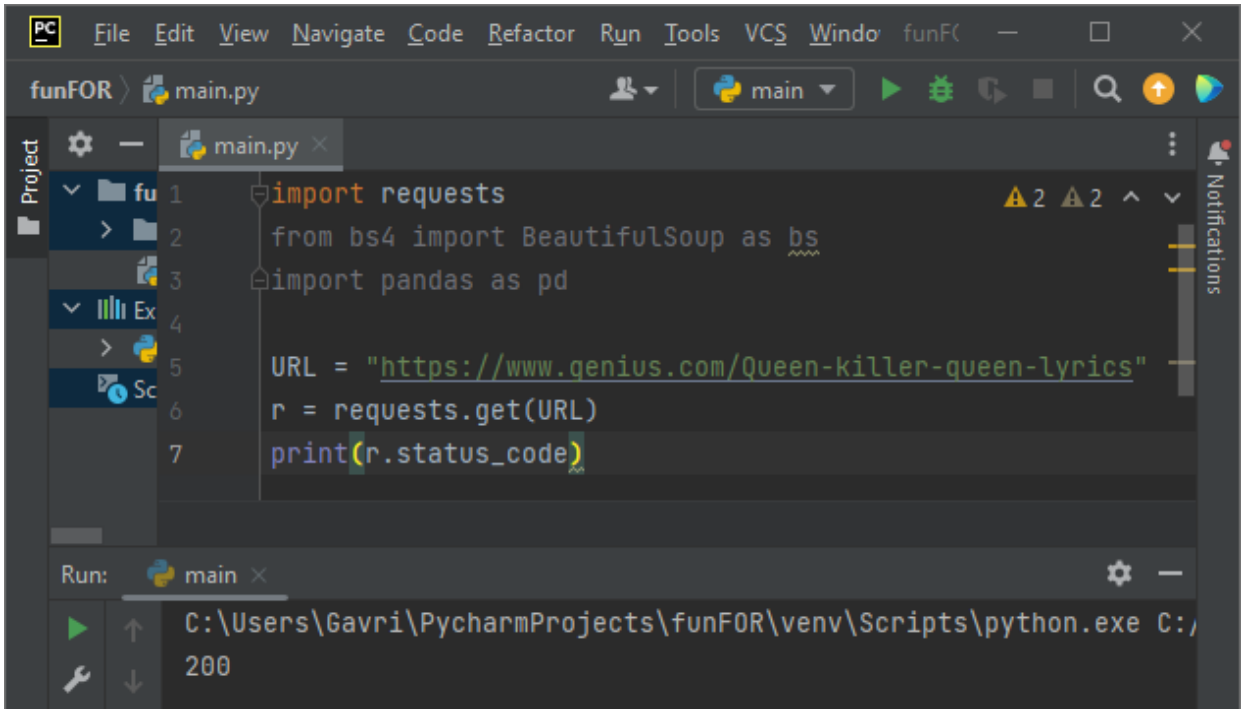

Рисунок 2 – Статус состояния HTTP

Ответ 200, значит подключение установлено. Можно получить HTML код страницы (рис. 3).

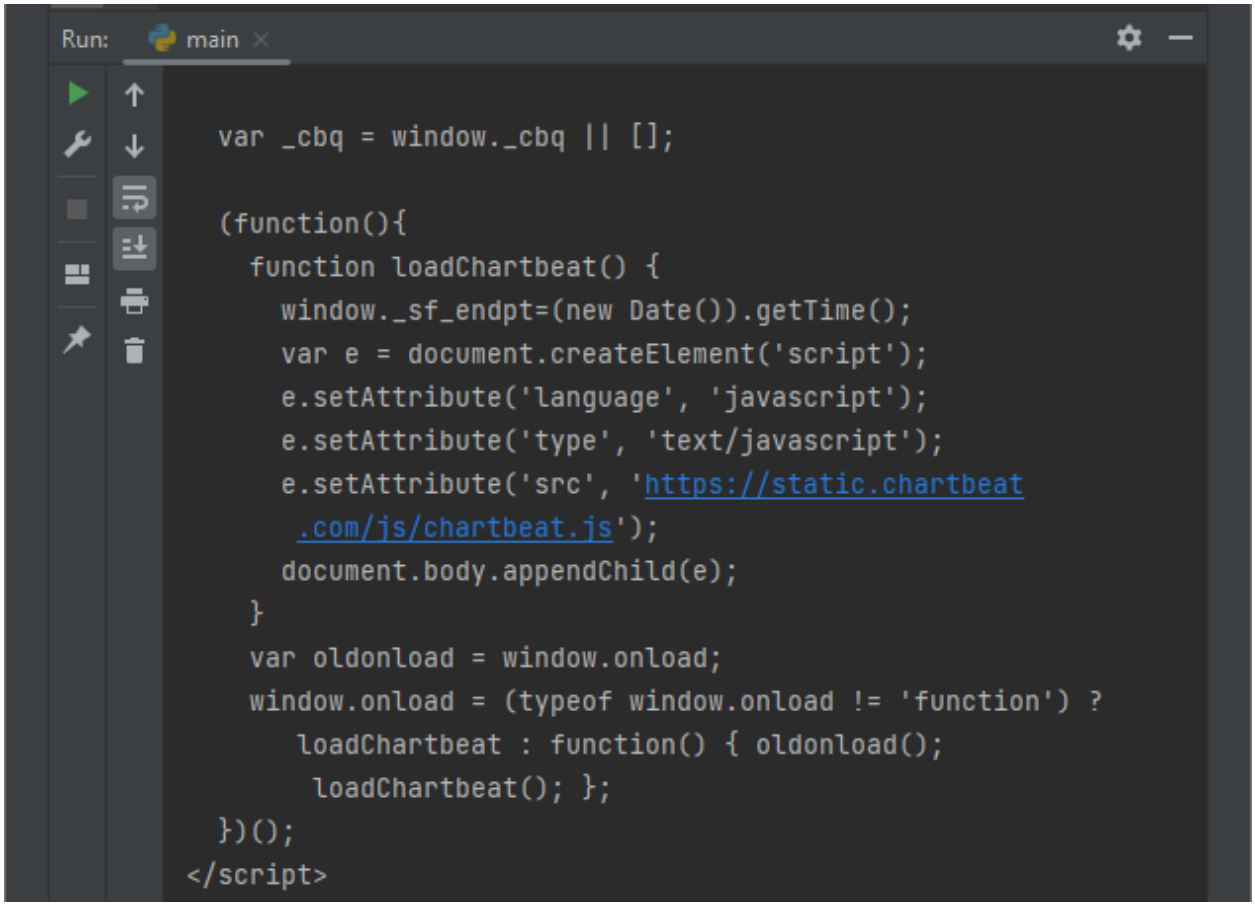

Рисунок 3 – Часть HTML кода страницы

Для того, чтобы разобраться, в каких тегах хранится искомый текст можно воспользоваться браузером. На примере песни группы Queen – Killer Queen и будет проводиться парсинг.

В браузере, для просмотра HTML кода используются инструменты разработчика. В Chrome их можно активировать, нажав F12. Исследуя страницу с текстом песни было установлено, что текст песни находится внутри тега «div» с конкретным классом (рис. 4).

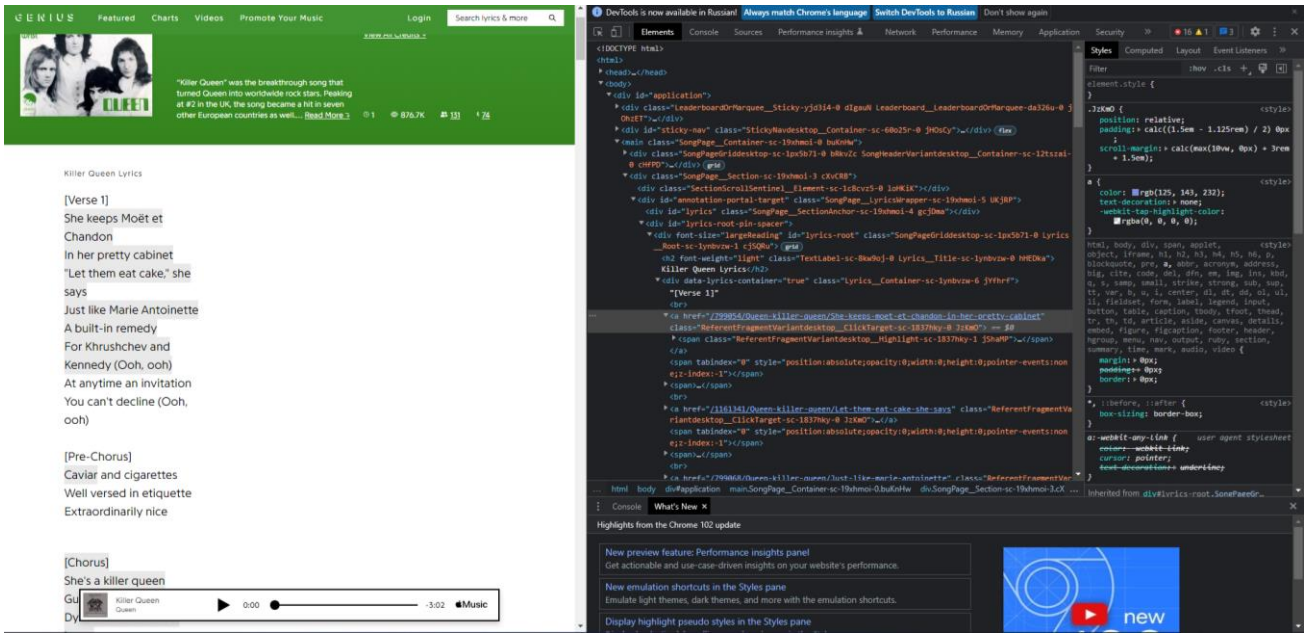

Рисунок 4 – Исследование HTML страницы

Когда известен тег и класс можно переходить к использованию библиотеки beautifulsoup4 (рис. 5).

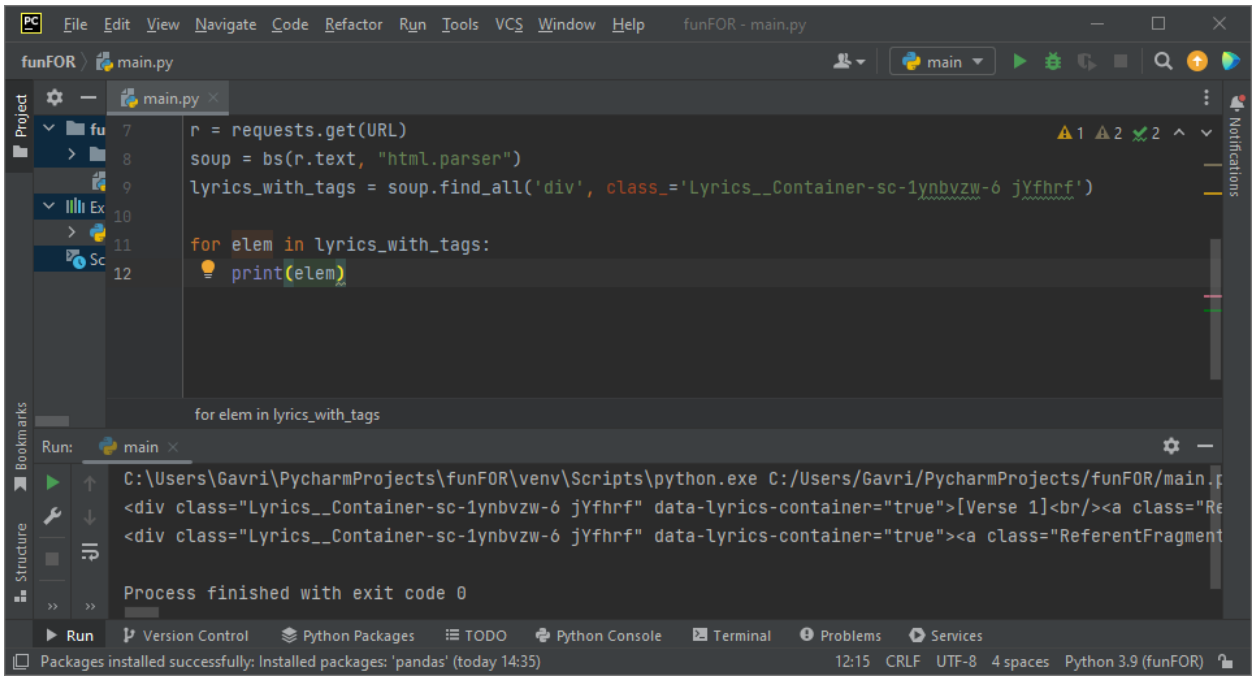

Рисунок 5 – Теги, содержащие текст песни

В 9 строке происходит поиск по тегу и классу, результат поиска сохраняется в объект «lyrics_with_tags», содержимое которого так же можно увидеть на рисунке. Осталось взять текст, разбить его на построчно, как в тексте песни. Реализовать такое становится возможно благодаря стандартным методам в работе со строками (функция tag_cleaner). Получившийся код на рисунке (рис. 6).

```
1  import requests
2  from bs4 import BeautifulSoup as bs
3  URL = "https://www.genius.com/Queen-killer-queen-lyrics"
4  r = requests.get(URL)
5  soup = bs(r.text, "html.parser")
6  lyrics_with_tags = soup.find_all('div', class_='Lyrics__Container-sc-1ynbvzw-6 jYfhrf')
7  lyrics = []
8
9
10 def tag_cleaner(tagged_str: str):
11     tagged_str = str(tagged_str)
12     tagged_str = tagged_str.replace("<br/>", "***")
13     flag = True
14     while flag:
15         start = tagged_str.find("<")
16         if start != -1:
17             stop = tagged_str.find(">")
18             tagged_str = tagged_str.replace(tagged_str[start:stop+1], "")
19         else:
20             flag = False
21     return list(tagged_str.split("***"))
22
23
24 for elem in lyrics_with_tags:
25     l = tag_cleaner(elem)
26     for e in l:
27         lyrics.append(e)
28 f = open("1.txt", "w", encoding='utf-8')
29 for string in lyrics:
30     f.write(string + "\n")
```

Рисунок 6 – Код программы

Результат работы программы сохраняется в файл (строки 28 - 30).

Следующим шагом является совершенствование программы для парсинга десяти самых популярных текстов с Genius (соответствующая страница есть на ресурсе).

С репозиторием проекта можно ознакомиться в [7]

3. Выводы

Таким образом, была создана программа для автоматизированного сбора данных с веб-ресурса Genius описанием её реализации.

Библиографический список

1. Genius. URL: https://genius.com/ (дата обращения: 11.06.2021).

2. Иванова С. В. Обзор библиотек для парсинга HTML на C# // Молодежь и системная модернизация страны: сборник научных статей 4-й Международной научной конференции студентов и молодых ученых, Курск, 21–22 мая 2019 года. Курск: Юго-Западный государственный университет, 2019. С. 75-77. URL: https://elibrary.ru/item.asp?id=38316326 (дата обращения: 11.06.2021).
3. Карабак И. И., Зорин К. А., Ажмухамедов И. М. Парсинг телеграм-каналов как элемент системы автоматизированного анализа информации, полученной из сети интернет // Прикаспийский журнал: управление и высокие технологии. 2022. №. 1 (57). С. 9-17. URL: https://cyberleninka.ru/article/n/parsing-telegram-kanalov-kak-element-sistemy-avtomatizirovannogo-analiza-informatsii-poluchennoy-iz-seti-internet (дата обращения: 11.06.2021).
4. Дубровина А. И. Парсинг, его последствия и методы предотвращения в целях защиты информации //Modern Science. 2020. №. 12-3. С. 230-233. URL: https://elibrary.ru/item.asp?id=44405062 (дата обращения: 11.06.2021).
5. Эшонкулов Х. И. Проблемы автоматизированного сбора информации // Вестник науки и образования. 2021. № 11-2(114). С. 38-41. URL: https://cyberleninka.ru/article/n/problemy-avtomatizirovannogo-sbora-informatsii (дата обращения: 11.06.2021).
6. Менщиков А. А., Комарова А. В., Гатчин Ю. А. Изучение поведения средств автоматизированного сбора информации с веб-ресурсов // Вопросы кибербезопасности. 2017. №. 3 (21). С. 49-54. URL: https://cyberleninka.ru/article/n/izuchenie-povedeniya-sredstv-avtomatizirovannogo-sbora-informatsii-s-veb-resursov (дата обращения: 11.06.2021).
7. GitHub. URL: https://github.com/Gavriilbolt/parser (дата обращения: 11.06.2021).