



**ПРОЕКТ «ИНСТРУМЕНТ ДЛЯ ПОИСКА  
ОТЗЫВОВ ПОЛЬЗОВАТЕЛЕЙ НА ЗАДАННЫЙ  
ТОВАР ИЛИ КОМПАНИЮ»**

Ученик 11б класса Егоров Никита



## ■ SWOT

### Сильные стороны:

- Опыт
- Помощь специалистов

### Слабые стороны:

- Небольшая база

### Возможности:

- Изучить работу веб-сайтов.
- Написать программу, работающую с данными сайта.

### Угрозы:

- Каждый сайт работает по-своему
- Программа может работать относительно неэффективно



```
File Edit Selection View Go Run Terminal Help
project.py - Visual Studio Code
project.py X
C:\Users> kibitka > Desktop > files > project > project.py
1 import requests
2 import lxml
3 import csv
4 import re
5 from bs4 import BeautifulSoup
6
7 URL = 'https://otzyvov.net/'
8 HEADERS = {'User-Agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/86.0.4240.75 Safari/537.36 Edg/86.0.622.11'}
9
10 def get_html(url, params = None):
11     r = requests.get(url, headers=HEADERS, params=params)
12     r.encoding = 'utf-8'
13     return r
14
15 def get_votes(soup):
16     votes = {}
17     votes['user'] = soup.find('div', class_='commRName').get_text()
18     votes['rating'] = len(soup.find('div', class_='commRTit').find_all('span', class_='starON'))
19     votes['date'] = soup.find('div', class_='commRDD').get_text()
20     votes['comm'] = soup.find('div', class_='otztxt-frist').get_text()
21     return votes
22
23 def get_otzyvov(html):
24     soup = BeautifulSoup(html, 'lxml')
25     url = soup.find('a', class_='lBx2BxPostL').get('href')
26     html = get_html(url)
27     soup = BeautifulSoup(html.text, 'lxml')
28
29     items = []
30     items.append({
31         'name': soup.find('div', class_='h4 text-center').get_text(),
32         'rating': len(soup.find('div', class_='rB1BxTtxtRating text-al-center-rB1').find_all('span', class_='starON')),
33         'count': soup.find('div', class_='rB1BxTtxtRatVote rB1BxTtxtRatVoteBold').get_text(strip=True),
34         'last vote': get_votes(soup),
35         'link': url
36     })
37     return items
38
39
40
41 def save_file(items, path):
42     with open(path, 'w', newline='') as file:
43         writer = csv.writer(file)
44         writer.writerow(items)
45
46 def main():
47     items = []
48     search = input()
49     req = {'s': search}
50     html = get_html(URL, req)
51
52     items.append(get_otzyvov(html.text))
53     save_file(items, 'items.csv')
54     print(items)
55
56 if __name__ == '__main__':
57     main()
```

