web-scraping

September 23, 2023

0.0.1 Web-scraping

[2]:
```
# %pip install requests beautifulsoup4
```

for finding tags - right click - see code - find tag

[ ]:
```

```

[11]:
```
import requests
import json
from bs4 import BeautifulSoup
import time

# Define the URL for the product's customer reviews page
url = "https://www.backmarket.es/es-es/p/iphone-13-128-gb-rosa-libre/
 ↪d1989098-648d-490e-a719-7c4b6ab8b1ae?
 ↪shopping=gmc&gclid=CjwKCAjwmbqoBhAgEiwACIjzEOW90Kp7IohNNjuJc3zyhFe2TholtFZA9_oeiEnQx_YIciyJ

# Define headers to mimic a web browser request
headers = {
    "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36␣
 ↪(KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.36"
}
time.sleep(3)
# Send an HTTP GET request to the URL with headers
response = requests.get(url, headers=headers)

# Check if the request was successful
if response.status_code == 200:
    # Parse the HTML content of the page using BeautifulSoup
    soup = BeautifulSoup(response.text, "html.parser")

    # Locate and extract customer reviews
    reviews = []
    review_elements = soup.find_all("div", class_="relative mt-7")

    for review_element in review_elements:
        review_text = review_element.find("span", class_="body-1-light␣
 ↪text-grey-500 whitespace-pre-line").text.strip()
        reviews.append(review_text)

    # Create a JSON object with the reviews
    review_data = {"reviews": reviews}

    # Save the JSON data to a file
    with open("backmarket_reviews.json", "w", encoding="utf-8") as json_file:
        json.dump(review_data, json_file, ensure_ascii=False, indent=4)

    print("Reviews have been scraped and saved to amazon_reviews.json.")
else:
    print(f"Failed to retrieve the page. Status code: {response.status_code}")
```

```
Reviews have been scraped and saved to amazon_reviews.json.
```

[ ]: