kumar Follow

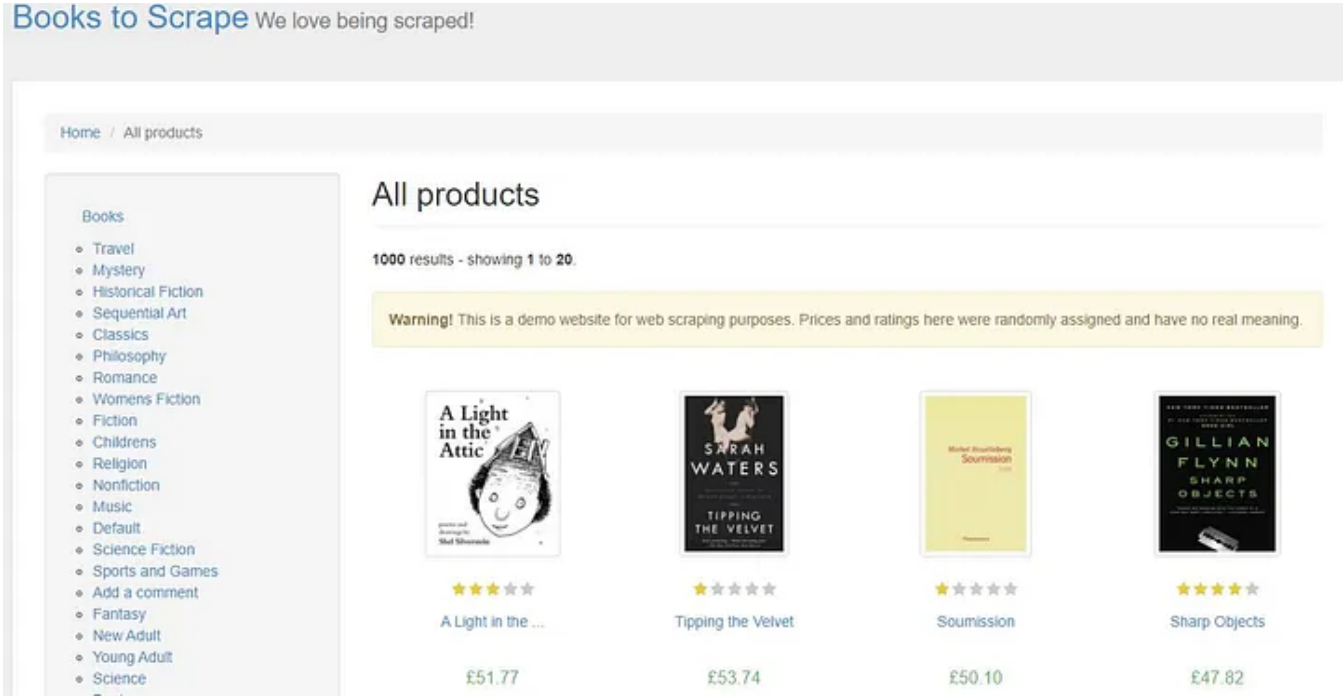
in read · Listen

Save

Web scraping Book data using Python

Python Project for Beginners

- Web scraping, web harvesting, or web data extraction is data scraping used for extracting data from websites. Web scraping a web page involves fetching it and extracting from it. Fetching is the downloading of a page (which a browser does when a user views a page). Once fetched, then extraction can take place. The content of a page may be parsed, searched, reformatted, its data copied into a spreadsheet or loaded into a database. Web scrapers typically take something out of a page, to make use of it for another purpose somewhere else.
- We are going to use Books to scrape site. A fictional bookstore that desperately wants to be scraped. It's a safe place for beginners learning web scraping and for developers validating their scraping technologies as well. Available at: https://books.toscrape.com/
- To do so, we are going to use tools like Python,Requests,BeautifulSoup,Pandas.



Books to scrape site

Outline:

From this site, we are going to grab the following information:

- book titles
- price
- stock availability
- link to get each book

After collecting the information, we are going to store it in a Pandas Data Frame and convert it to CSV file.

Importing the required libraries and dependencies

Downloading the webpage using requests

Use Beautiful Soup to parse and extract the information

NOTE * *Beautiful Soup* is a python library which pulls data out of HTML and XML documents. It creates a parse tree for parsed pages, which can be used to pull out extract data from HTML. * It helps programmers to save hours or days of work.

Grabbing book titles

In this module, we are going to create a helper function `get_book_titles` to grab the book titles from the document.

`get_book_titles(doc)` will give you the titles of the books like the image shown below.

```
In [ ]: def get_book_titles(doc):
            Book_title_tags = doc.find_all('h3')
            Book_titles = []
            for tags in Book_title_tags:
                Book_titles.append(tags.text)
            return Book_titles
```

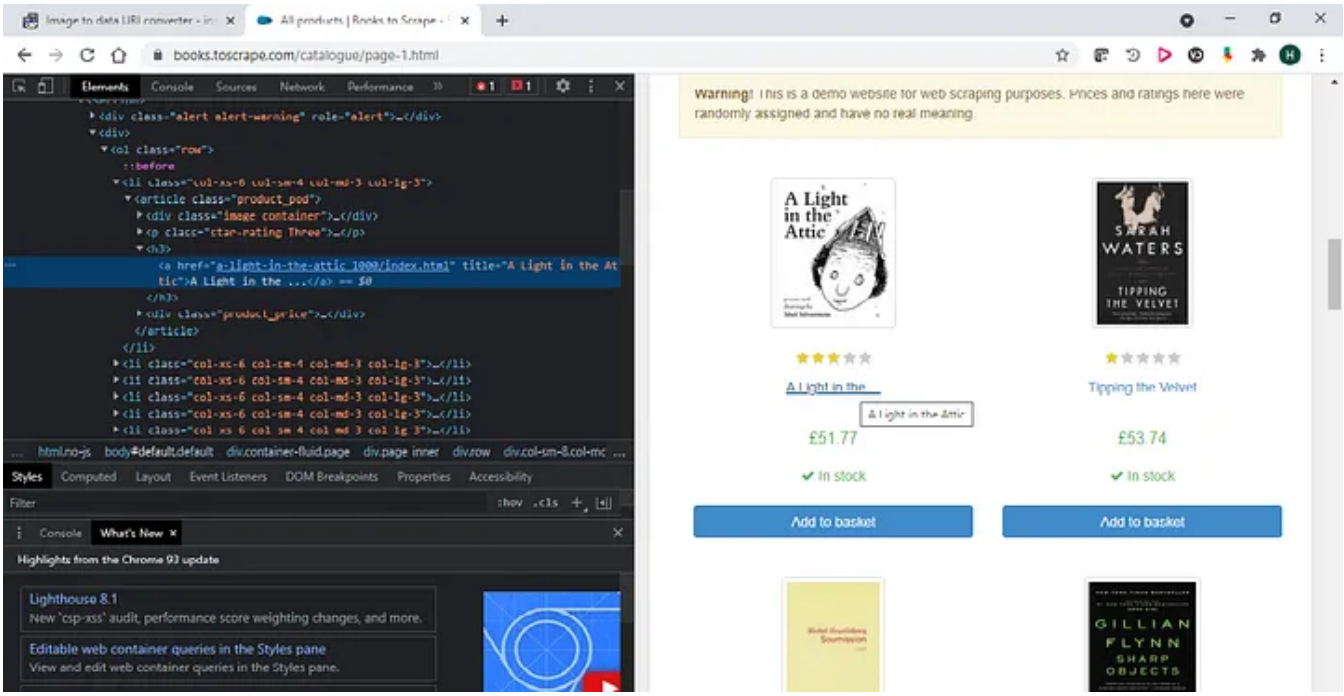
```
In [94]: get_book_titles(doc)
```

```
Out[94]: ['A Light in the ...',
          'Tipping the Velvet',
          'Soumission',
          'Sharp Objects',
          'Sapiens: A Brief History ...',
          'The Requiem Red',
          'The Dirty Little Secrets ...',
          'The Coming Woman: A ...',
          'The Boys in the ...',
          'The Black Maria',
          'Starving Hearts (Triangular Trade ...',
          "Shakespeare's Sonnets",
          'Set Me Free',
          "Scott Pilgrim's Precious Little ...",
          'Rip it Up and ...',
          'Our Band Could Be ...',
          'Olio',
          'Mesaerion: The Best Science ...',
```

OUTPUT : Book Titles

Using Inspect element to get the location of needed information.

`get_book_titles` grabs the text from a tag within the h3 tag.



Grabbing book prices

In this module, we are going to create a helper function `get_book_price` to grab the price of each book from the document.

`get_book_price(doc)` will give you the titles of the books like the image shown below.

```
In [32]: def get_book_price(doc):
             Book_price_tags = doc.find_all('p', class_ = 'price_color')
             Book_price = []
             for tags in Book_price_tags:
                 Book_price.append(tags.text.replace('Â',''))
             return Book_price
```

```
In [101]: get_book_price(doc)

Out[101]: ['£51.77',
           '£53.74',
           '£50.10',
           '£47.82',
           '£54.23',
           '£22.65',
           '£33.34',
           '£17.93',
           '£22.60',
           '£52.15',
           '£13.99',
           '£20.66',
           '£17.46',
           '£52.29',
           '£35.02',
           '£57.25',
           '£23.88',
           '£37.59',
```

OUTPUT : Book Prices

Grabbing stock availability

In this module, we are going to create a helper function `get_stock_availability` to grab the stock availability from the document and store in the variable `Book_stock` .

Grabbing links for each book

In this module, we are going to create a helper function `get_book_url` to grab the links for each book.

`get_book_url(Book_title_tags)` will give you the links for each book like the image shown below.

```
In [34]: def get_book_url(Book_title_tags):
             Book_url = []
             for article in Book_title_tags:
                 for link in article.find_all('a', href = True):
                     url = link['href']
                     links = 'https://books.toscrape.com/' + url
                     if links not in Book_url:
                         Book_url.append(links)
             return Book_url

In [102]: get_book_url(Book_title_tags)

Out[102]: ['https://books.toscrape.com/catalogue/a-light-in-the-attic_1000/index.html',
           'https://books.toscrape.com/catalogue/tipping-the-velvet_999/index.html',
           'https://books.toscrape.com/catalogue/soumission_998/index.html',
           'https://books.toscrape.com/catalogue/sharp-objects_997/index.html',
           'https://books.toscrape.com/catalogue/sapiens-a-brief-history-of-humankind_996/index.html',
           'https://books.toscrape.com/catalogue/the-requiem-red_995/index.html',
           'https://books.toscrape.com/catalogue/the-dirty-little-secrets-of-getting-your-dream-job_994/index.html',
           'https://books.toscrape.com/catalogue/the-coming-woman-a-novel-based-on-the-life-of-the-infamous-feminist-victoria-woodhull_99
          3/index.html',
           'https://books.toscrape.com/catalogue/the-boys-in-the-boat-nine-americans-and-their-epic-quest-for-gold-at-the-1936-berlin-oly
          mpics_992/index.html',
           'https://books.toscrape.com/catalogue/the-black-maria_991/index.html',
           'https://books.toscrape.com/catalogue/starving-hearts-triangular-trade-trilogy-1_990/index.html',
           'https://books.toscrape.com/catalogue/shakespeares-sonnets_989/index.html',
```

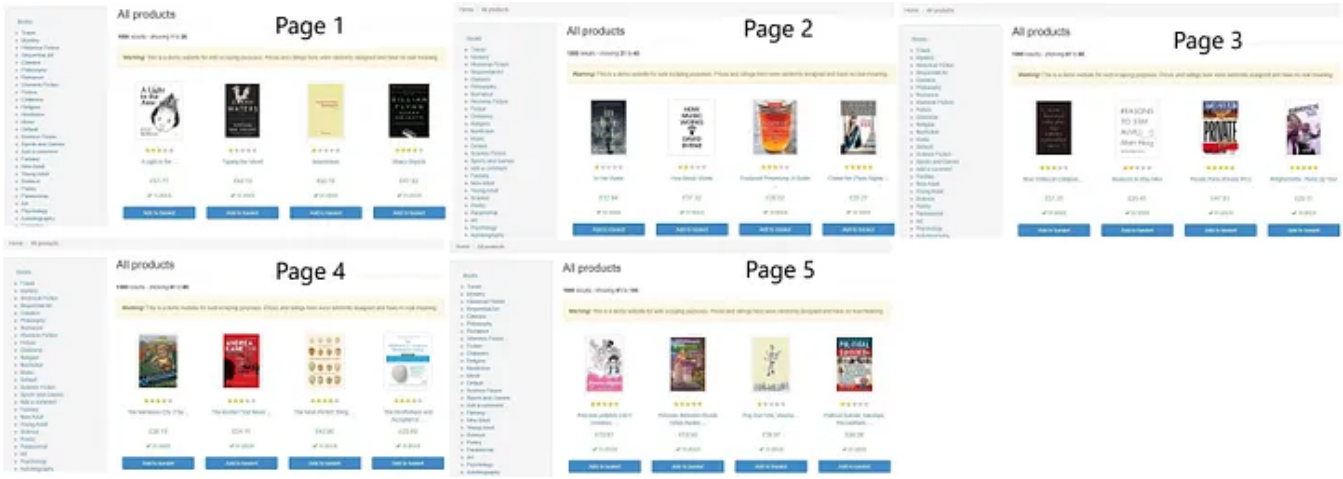
OUTPUT: Links for each book

Collecting data from multiple pages and store in Pandas DataFrame

- So far, we have collected information from the single page (or) the first page of our website. `scrape_single_page` collects the information from the first page.

- In this section, we are going to create function `scrape_multiple_pages` to collect the information from multiple pages.
- Store the collected information into Pandas Data Frame.
- In function `scrape_multiple_pages` , we have used 'n' to collect from multiple pages, here we are going to collect from *5* pages and so $n = 5$.

The image shows the first five pages, from where our information is to be scraped.



First five pages that are scraped.

- A variable `book_dict1` is created to store the information as dictionary.
- Scraped information is stored into Pandas DataFrame.

- The image shows the DataFrame containing *100 rows and 4 columns* of data.

```
In [91]: scrape_multiple_pages(5)
```

Out[91]:

	TITLE	PRICE	STOCK AVAILABILTY	URL
0	A Light in the ...	£51.77	In stock	https://books.toscrape.com/a-light-in-the-atti...
1	Tipping the Velvet	£53.74	In stock	https://books.toscrape.com/tipping-the-velvet_...
2	Soumission	£50.10	In stock	https://books.toscrape.com/soumission_998/inde...
3	Sharp Objects	£47.82	In stock	https://books.toscrape.com/sharp-objects_997/i...
4	Sapiens: A Brief History ...	£54.23	In stock	https://books.toscrape.com/sapiens-a-brief-his...
...	...	...	...	...
95	Lumberjanes Vol. 3: A ...	£19.92	In stock	https://books.toscrape.com/lumberjanes-vol-3-a...
96	Layered: Baking, Building, and ...	£40.11	In stock	https://books.toscrape.com/layered-baking-buil...
97	Judo: Seven Steps to ...	£53.90	In stock	https://books.toscrape.com/judo-seven-steps-to...
98	Join	£35.67	In stock	https://books.toscrape.com/join_902/index.html
99	In the Country We ...	£22.00	In stock	https://books.toscrape.com/in-the-country-we-l...

100 rows × 4 columns

Creating a CSV file

- This section involves.,
- conversion of information stored to a CSV file named SCB.csv.
- Serial numbers or the index values are removed using index = None

The image shows the created csv file ***SCB.csv***

```
TITLE,PRICE,STOCK AVAILABILTY,URL
A Light in the ...,£51.77,In stock,https://books.toscrape.com/a-light-in-the-attic_1000/index.html
Tipping the Velvet,£53.74,In stock,https://books.toscrape.com/tipping-the-velvet_999/index.html
Soumission,£50.10,In stock,https://books.toscrape.com/soumission_998/index.html
Sharp Objects,£47.82,In stock,https://books.toscrape.com/sharp-objects_997/index.html
Sapiens: A Brief History ...,£54.23,In stock,https://books.toscrape.com/sapiens-a-brief-history-of-humankind_996/index.html
The Requiem Red,£22.65,In stock,https://books.toscrape.com/the-requiem-red_995/index.html
The Dirty Little Secrets ...,£33.34,In stock,https://books.toscrape.com/the-dirty-little-secrets-of-getting-your-dream-job_994/index.html
The Coming Woman: A ...,£17.93,In stock,https://books.toscrape.com/the-coming-woman-a-novel-based-on-the-life-of-the-infamous-feminist-victoria-woodhull_993/index.html
The Boys in the ...,£22.60,In stock,https://books.toscrape.com/the-boys-in-the-boat-nine-americans-and-their-epic-quest-for-gold-at-the-1936-berlin-olympics_992/index.html
The Black Maria,£52.15,In stock,https://books.toscrape.com/the-black-maria_991/index.html
Starving Hearts (Triangular Trade ...,£13.99,In stock,https://books.toscrape.com/starving-hearts-triangular-trade-trilogy-1_990/index.html
Shakespeare's Sonnets,£20.66,In stock,https://books.toscrape.com/shakespeares-sonnets_989/index.html
Set Me Free,£17.46,In stock,https://books.toscrape.com/set-me-free_988/index.html
Scott Pilgrim's Precious Little ...,£52.29,In stock,https://books.toscrape.com/scott-pilgrims-precious-little-life-scott-pilgrim-1_987/index.html
Rip it Up and ...,£35.02,In stock,https://books.toscrape.com/rip-it-up-and-start-again_986/index.html
Our Band Could Be ...,£57.25,In stock,https://books.toscrape.com/our-band-could-be-your-life-scenes-from-the-american-indie-underground-1981-1991_985/index.html
Olio,£23.88,In stock,https://books.toscrape.com/olio_984/index.html
Mesaerion: The Best Science ...,£37.59,In stock,https://books.toscrape.com/mesaerion-the-best-science-fiction-stories-1800-1849_983/index.html
Libertarianism for Beginners,£51.33,In stock,https://books.toscrape.com/libertarianism-for-beginners_982/index.html
It's Only the Himalayas,£45.17,In stock,https://books.toscrape.com/its-only-the-himalayas_981/index.html
In Her Wake,£12.84,In stock,https://books.toscrape.com/in-her-wake_980/index.html
How Music Works,£37.32,In stock,https://books.toscrape.com/how-music-works_979/index.html
```

Created CSV file

Downloaded CSV file in Excel

	A	B	C	D
1	TITLE	PRICE	STOCK AVAILABILTY	URL
2	A Light in the ...	Â£51.77	In stock	https://books.toscrape.com/a-light-in-the-attic_1000/index.html
3	Tipping the Velvet	Â£53.74	In stock	https://books.toscrape.com/tipping-the-velvet_999/index.html
4	Soumission	Â£50.10	In stock	https://books.toscrape.com/soumission_998/index.html
5	Sharp Objects	Â£47.82	In stock	https://books.toscrape.com/sharp-objects_997/index.html
6	Sapiens: A Brief History ...	Â£54.23	In stock	https://books.toscrape.com/sapiens-a-brief-history-of-humankind_996/index.html
7	The Requiem Red	Â£22.65	In stock	https://books.toscrape.com/the-requiem-red_995/index.html
8	The Dirty Little Secrets ...	Â£33.34	In stock	https://books.toscrape.com/the-dirty-little-secrets-of-getting-your-dream-job_994/index.html
9	The Coming Woman: A ...	Â£17.93	In stock	https://books.toscrape.com/the-coming-woman-a-novel-based-on-the-life-of-the-infamous-feminist-victoria-woodhull_993/index.html
				https://books.toscrape.com/the-boys-in-the-boat-nine-americans-and-their-epic-quest-for-gold-at-the-1936-

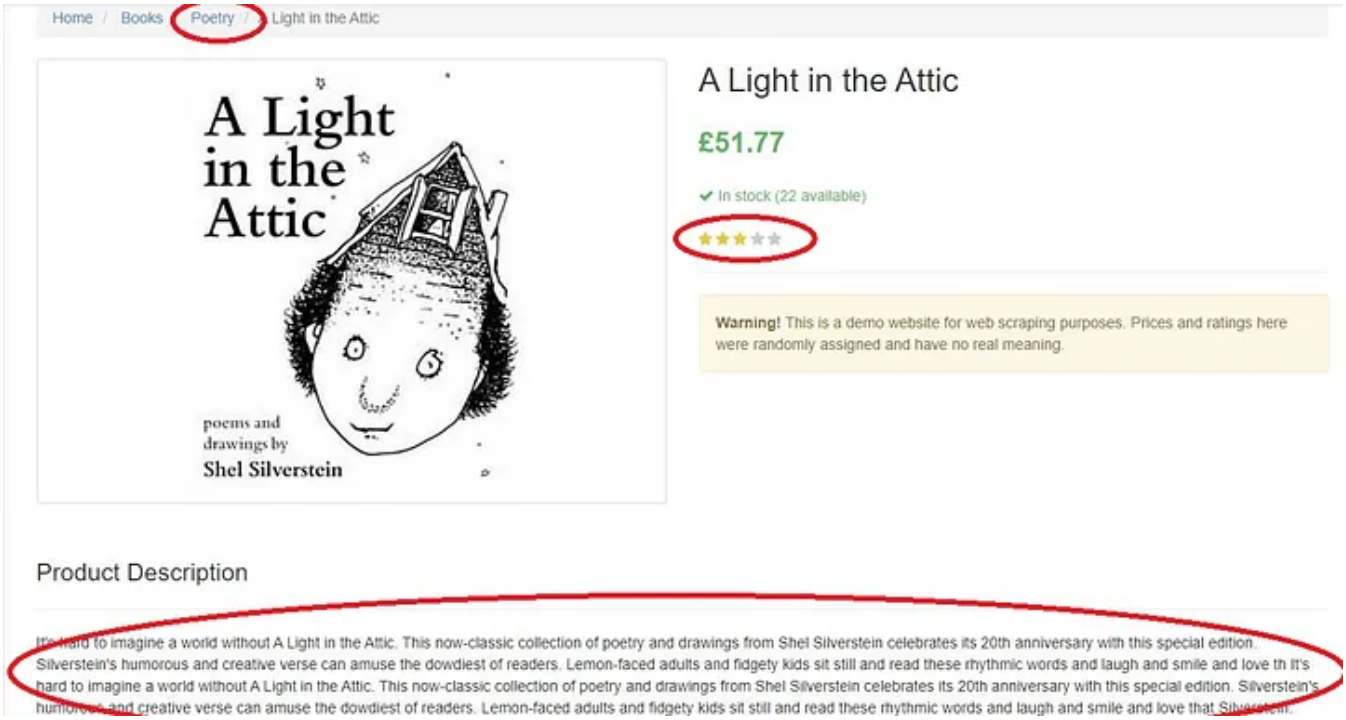
Output: Excel file

Summary and Future Scope

- In this Project, we used the most popular web-scraping package Beautiful Soup, which creates a parse tree that can be used to extract data from HTML on a website. Beautiful soup also has multiple features for navigation, searching, and modifying these parse trees. In addition to quickly grabbing the information we

need, you can create the program within few lines of code without having to write dozens of lines of code.

- From the : https://books.toscrape.com/ site, we have scraped data such as., book titles, price, stock availability, link to get each book, using helper functions.
- After which, we have put the scraped contents in ***Pandas DataFrame***.
- Finally, converted it to a CSV file, ***SCB.csv***.



First page of Books to scrape site

Refer the image above. Following the same method we used in this project, we can go further and collect more information this time book-wise (like the image) such as:

- star ratings
- description of the book
- genre
- cover page etc

References

- Here are some references, which will be useful in web scraping using python.

- Documentation of BeautifulSoup. *https://www.crummy.com/software/BeautifulSoup/bs4/doc/*
- A guide to Requests library *https://realpython.com/python-requests/*
- Pandas tutorial *https://pandas.pydata.org/pandas-docs/stable/user_guide/10min.html*
- Writing functions in python *https://www.datacamp.com/community/tutorials/functions-python-tutorial*
- Web scraping tutorials *https://www.youtube.com/watch?v=ng2o98k983k* *https://www.youtube.com/watch?v=RKsLLG-bzEY&t=8458s*

Complete code:
https://github.com/meenakshiravi1/Webscraping/blob/main/Python%20Webscraping%20Project.

Web Scraping Python 3