Written Assignment

Web Scrapping

AUTHOR(s): Payyappilly Tomy, Vinu

PROGRAM: Data Science

MODULE: Introduction to DS

TYPE OF ASSIGNMENT:
☐ Pre-Module
☒ Post-Module
☐ Other

IMPORTANT NOTES

Plagiarism: All student work is checked for plagiarism upon submission. To reduce the plagiarism detection value, please do not include the assignment instructions in the submitted document.

File name:

Sometimes file names are so long, we cannot save the document. Please name your documents for both individual and group works like this: **"Surname_PRE/POST"**

WU EXECUTIVE ACADEMY
WIRTSCHAFTSUNIVERSITÄT
WIEN VIENNA UNIVERSITY OF ECONOMICS
AND BUSINESS

Welthandelsplatz 1, Building EA, 1020 Vienna, Austria
T +43-1-313 36-4816, F +43-1-313 36-904816
executiveacademy@wu.ac.at
executiveacademy.at

Bank: UniCredit Bank Austria AG, BLZ: 11000
Account No.: 01364 971 000
SWIFT/BIC: BKAUATWW, IBAN: AT45 1100 0013 6497 1000
VAT No.: ATU37694107, Place of jurisdiction Vienna

WebScrapping Using Beautiful Soap and Python Lib.

I have tried using Python request library to get automated Text scrapping. But most of the sites I have planned is blocking such API request from scripts. I am getting either access denied or API blocking error message.

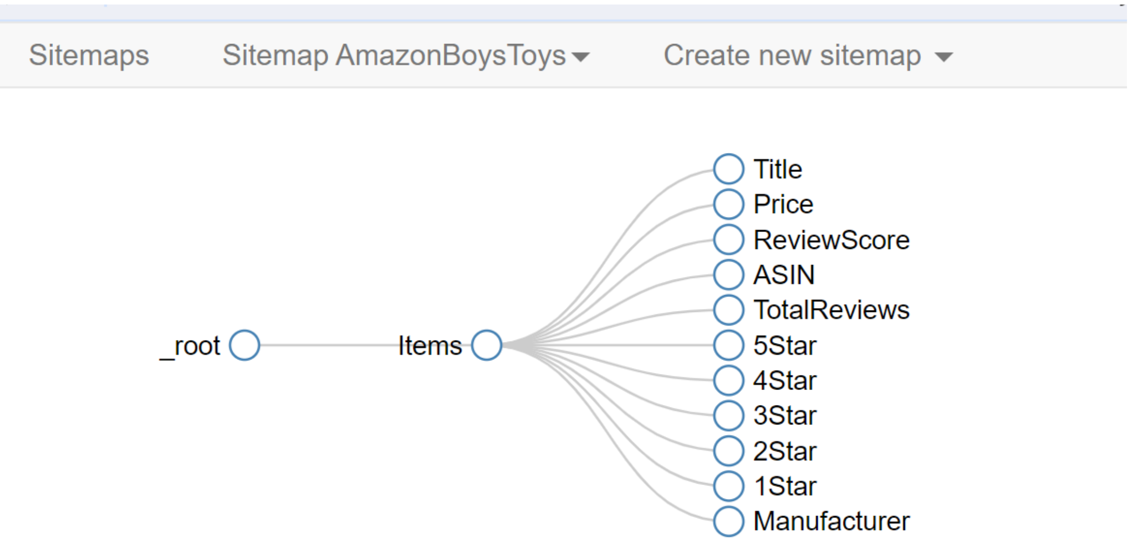
I have used below script :

```
import requests

from bs4 import BeautifulSoup

r = requests.get("https://www.smythstoys.com/at/de-at/spielzeug/action-spielzeug/actionfiguren/roblox-actionfiguren/roblox-adventskalender/p/203665")

type(r)

r.content

soup = BeautifulSoup(r.content, "lxml")

# finds first tag where type is h1 and class name is "fn"

element = soup.find("h1", class_="fn")

element
```

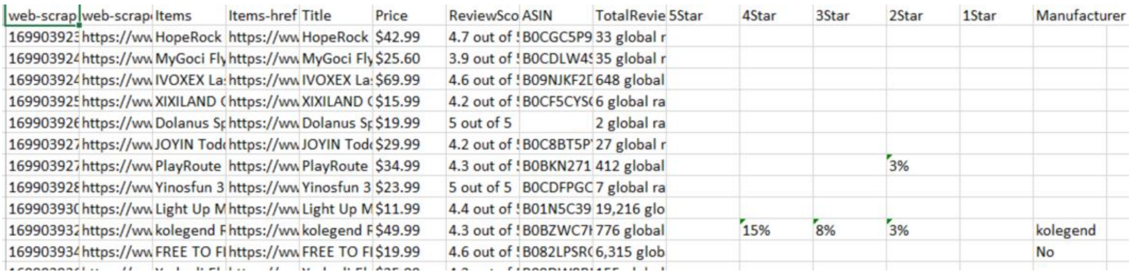
Then I decided to use Browser plugin to avoid access issues.

WebScrapping using Chrome plugin

I have used webscrapper.io chrome plug in and used Amazon site targeted on Boys toys search result to limit the analysis scope. I have created below Selector map to scrap the data with pagination applied.



It scrapped more than 300 results in csv format. I was interested to find the Rating scores and Manufacture and Price comparison to rating score. It is observed that Not all star rating available and Manufacture information is also missing many records. So it needed a data cleaning and preparation task. I have used Python panda DF for the cleansing task.



web-scrap	web-scrap	Items	Items-href	Title	Price	ReviewSco	ASIN	TotalRevie	5Star	4Star	3Star	2Star	1Star	Manufacturer
169903923	https://ww	HopeRock	https://ww	HopeRock	$42.99	4.7 out of 5	B0CGC5P9	33 global r						
169903924	https://ww	MyGoci Fly	https://ww	MyGoci Fly	$25.60	3.9 out of 5	B0CDLW4	35 global r						
169903924	https://ww	IVOXEX La	https://ww	IVOXEX La	$69.99	4.6 out of 5	B09NJKF2D	648 global						
169903925	https://ww	XIXILAND (	https://ww	XIXILAND (	$15.99	4.2 out of 5	B0CF5CYS	6 global ra						
169903926	https://ww	Dolanus Sp	https://ww	Dolanus Sp	$19.99	5 out of 5		2 global ra						
169903927	https://ww	JOYIN Tod	https://ww	JOYIN Tod	$29.99	4.2 out of 5	B0C8BT5P	27 global r						
169903927	https://ww	PlayRoute	https://ww	PlayRoute	$34.99	4.3 out of 5	B0BKN271	412 global				3%		
169903928	https://ww	Yinosfun 3	https://ww	Yinosfun 3	$23.99	5 out of 5	B0CDFPGC	7 global ra						
169903930	https://ww	Light Up M	https://ww	Light Up M	$11.99	4.4 out of 5	B01N5C39	19,216 glo						
169903932	https://ww	kolegend F	https://ww	kolegend F	$49.99	4.3 out of 5	B0BZWC7	776 global		15%	8%	3%		kolegend
169903934	https://ww	FREE TO Fl	https://ww	FREE TO Fl	$19.99	4.6 out of 5	B082LPSR	6,315 glob						No

I have used below script to clean the CSV

import csv

import pandas as pd

import matplotlib.pyplot as plt

file = "./data/AmazonBoysToys.csv"

df = pd.read_csv(file)

print(df.info())

```
## drop duplicates

df.drop_duplicates(inplace = True)

## fill emplty  values  and correct incorrect values for analysis

df["Manufacturer"].fillna("NoData", inplace = True)

df["ASIN"].fillna("NoData", inplace = True)

df["TotalReviews"].fillna("NoData", inplace = True)

df["TotalReviews"].fillna("NoData", inplace = True)

df['TotalReviews'] = df['TotalReviews'].str.replace('[global ratings]', '')

df['ReviewScore'] = df['ReviewScore'].str.replace('[out of 5]', '')

df["ReviewScore"].fillna("0", inplace = True)

df['Price'] = df['Price'].str.replace('[$]', '')

df["Price"].fillna("0", inplace = True)


## fill emplty  star rating values

df["1Star"].fillna("0%", inplace = True)

df["2Star"].fillna("0%", inplace = True)

df["3Star"].fillna("0%", inplace = True)

df["4Star"].fillna("0%", inplace = True)

df["5Star"].fillna("0%", inplace = True)

## Removes special chars in Title column

df['Title'] = df['Title'].str.replace('[#,@,&,|,:,-]', ' ')

df['Items'] = df['Items'].str.replace('[#,@,&,|,:,-]', ' ')

## conevrt to Lower case

df['Title'] = df['Title'].str.lower()

df['Items'] = df['Items'].str.lower()

## trim end of string

df['Title'] = df['Title'].str.strip()
```

```
df['Items'] = df['Items'].str.strip()

df.to_csv('./data/AmazonBoysToysCleaned.csv')

print(df.info())

df.head(10)
```