***WEB SCRAPING USING PYTHON***

We will scrape the amazon data using the beautiful soup python library which is used for web scraping purposes.we will use the google colab platform to do this task. The prerequistics for this task are as follows.

Make sure you have python 3 or higher install in you system…

We need to install the following…

Pip install bs4

Pip install requests

Pip install pandas

Then do the following imports…

from bs4 import BeautifulSoup

import requests

import pandas as pd

Then visit the Amazon website and search for the product “baby wipes”

Then copy the url.

THEN VISIT WHATISMYBROWER.com to get the user agent for the request headers for the Amazon website…

from bs4 import BeautifulSoup

import requests

import pandas as pd

URL="https://www.amazon.com/s?k=baby+wipes&crid=3BDUNHTQ355S&sprefix=baby+wipes%2Caps%2C270&ref=nb\_sb\_noss\_1"

#HEADERS FOR REQUEST

HEADERS=({'User-Agent':'','Accept-Language':'en-US,en;q=0.5'})

PASTE YOUR USER AGENT FROM THE BROWSER WHATISMYBROWSER.com

My user agent…

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36

HEADERS=({'User-Agent':'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/123.0.0.0 Safari/537.36','Accept-Language':'en-US,en;q=0.5'})

from bs4 import BeautifulSoup

import requests

import pandas as pd

URL="https://www.amazon.com/s?k=baby+wipes&crid=3BDUNHTQ355S&sprefix=baby+wipes%2Caps%2C270&ref=nb\_sb\_noss\_1"

#HEADERS FOR REQUEST

HEADERS=({'User-Agent':'','Accept-Language':'en-US,en;q=0.5'})

webpage=requests.get(URL,headers=HEADERS)

THEN TYPE :

WEBPAGE : you will get a response <200>

And when you will type webpage.content then the whole content will be displayed.

type(webpage.content)

#soup object containing all data for this we will use beautiful soup.

soup=BeautifulSoup(webpage.content,’html-parser’)

from bs4 import BeautifulSoup

import requests

import pandas as pd

URL="https://www.amazon.com/s?k=baby+wipes&crid=3BDUNHTQ355S&sprefix=baby+wipes%2Caps%2C270&ref=nb\_sb\_noss\_1"

#HEADERS FOR REQUEST

HEADERS=({'User-Agent':'','Accept-Language':'en-US,en;q=0.5'})

webpage=requests.get(URL,headers=HEADERS)

type(webpage.content)

soup=BeautifulSoup(webpage.content,'html.parser')

soup

from bs4 import BeautifulSoup

import requests

import pandas as pd

URL="https://www.amazon.com/s?k=baby+wipes&crid=3BDUNHTQ355S&sprefix=baby+wipes%2Caps%2C270&ref=nb\_sb\_noss\_1"

#HEADERS FOR REQUEST

HEADERS=({'User-Agent':'','Accept-Language':'en-US,en;q=0.5'})

webpage=requests.get(URL,headers=HEADERS)

type(webpage.content)

soup=BeautifulSoup(webpage.content,'html.parser')

soup

from bs4 import BeautifulSoup

import requests

import pandas as pd

URL="https://www.amazon.com/s?k=baby+wipes&crid=3BDUNHTQ355S&sprefix=baby+wipes%2Caps%2C270&ref=nb\_sb\_noss\_1"

#HEADERS FOR REQUEST

HEADERS=({'User-Agent':'','Accept-Language':'en-US,en;q=0.5'})

webpage=requests.get(URL,headers=HEADERS)

type(webpage.content)

soup=BeautifulSoup(webpage.content,'html.parser')

link=soup.find\_all("a",attrs={'class':'a-link-normal s-underline-text s-underline-link-text s-link-style a-text-normal'})

link

------------------------------------------------------------------------------------------------------------------------------------------

Link1=Link[0].get(‘href’)

Product\_list=<https://amazon.com> +Link1

Print(Product\_list)

**WEB SCRAPING OF AN AMAZON PRODUCT**

We will scrape an amazon product & convert to the desired excel format using the bs4 , openpyxl and request library.

CODE IN PYTHON:

from bs4 import BeautifulSoup

import requests

import pandas as pd

def scrape1\_amazon\_reviews(url):

    # Send a GET request to the URL and store the response

    response1 = requests.get(url, headers={'Accept-Language': 'en-US,en;q=0.5'})

    # Parse the HTML content of the page

    soup = BeautifulSoup(response1.content, 'html.parser')

    # Find all review sections

    reviews = soup.find\_all('div', class\_='a-section review aok-relative')

    # Initialize lists to store review data

    review\_data = []

    # Iterate over each review

    for review in reviews:

        # Initialize a dictionary to store review details

        review\_dict = {}

        # Find reviewer name

        reviewer\_name = review.find('span', class\_='a-profile-name').get\_text(strip=True)

        review\_dict['Reviewer Name'] = reviewer\_name

        # Find review date

        review\_date = review.find('span', class\_='a-size-base a-color-secondary review-date').get\_text(strip=True)

        review\_dict['Date'] = review\_date

        # Find review body

        review\_body = review.find('span', class\_='a-size-base review-text review-text-content').get\_text(strip=True)

        review\_dict['Comment'] = review\_body

        # Find review rating

        review\_rating = review.find('i', class\_='a-icon-star').get\_text(strip=True).split(' ')[0]

        review\_dict['Rating'] = review\_rating

        # Append review details to the list

        review\_data.append(review\_dict)

    # Create a DataFrame from the list of dictionaries

    df\_reviews = pd.DataFrame(review\_data)

    return df\_reviews

# URL of the Amazon product reviews page

URL = "https://www.amazon.co.jp/-/en/Ferrero-Rocher-T-30-Chocolate-ea/product-reviews/B00600QFA8/ref=cm\_cr\_arp\_d\_viewopt\_sr?filterByStar=five\_star&pageNumber=1"

# Scrape reviews from the provided URL

df\_reviews = scrape1\_amazon\_reviews(URL)

# Display the DataFrame

print(df\_reviews)

# Save the DataFrame to an Excel file

df\_reviews.to\_excel('amazon\_reviews120.xlsx', index=False)

EXCEL OUTPUT:

