**Web Crawler Assignment:**

As per you requirements to scrape an Amazon website for any product and stored in a Excel.

**Sample Input:** Given by User

Enter the Product Name: Mobile (any Product Name)

Enter the Page No to stop: 20

**Source Code:** any\_amazon.py

#import functions

import requests

from bs4 import BeautifulSoup

import pandas as pd

# Storing Output in the Dataframe

df1 = pd.DataFrame()

# Getting Product Name and No. of Pages by input from User

product = input("Enter the Product Name: ")

End\_Page\_No = int(input("Enter the Page No to stop: "))

# Looping No. of pages for particular product

for i in range(1,End\_Page\_No+1):

    url = f'https://www.amazon.in/s?k={product}&page={i}&qid=1650442227&ref=sr\_pg\_{i}'

    print('Crawling Page - '+str(i))

    headers = {'content-type': 'text/html;charset=UTF-8',

    'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9',

    'user-agent': 'Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896.75 Safari/537.36'}

    # Getting Response for the product

    response = requests.get(url = url,headers = headers)

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

    div\_tag = soup.find\_all('div',{'class' : 's-result-item s-asin sg-col-0-of-12 sg-col-16-of-20 sg-col s-widget-spacing-small sg-col-12-of-16'})

    # Assigning product detials based on Field Name

    Fields = {}

    for div in div\_tag:

        Fields['Source'] = 'Amazon'

        Fields['Product\_Name'] = div.find\_all('h2')[0].get\_text()

        Fields['Link'] = "https://www.amazon.in"+str(div.find\_all('a',{'class' : 'a-link-normal s-underline-text s-underline-link-text s-link-style a-text-normal'})[0]['href'])

        if div.find\_all('span',{'class' : 'a-icon-alt'}):

            Fields['Star'] = (div.find\_all('span',{'class' : 'a-icon-alt'})[0].get\_text()).replace(' out of 5 stars','')

        if div.find\_all('span',{'class' : 'a-size-base s-underline-text'}):

            Fields['Reviews'] = div.find\_all('span',{'class' : 'a-size-base s-underline-text'})[0].get\_text().strip()

        if div.find\_all('a',{'class' : 'a-link-normal s-underline-text s-underline-link-text s-link-style'}):

            if (div.find\_all('a',{'class' : 'a-link-normal s-underline-text s-underline-link-text s-link-style'})[0]['href']) == '#':

                Fields['Review\_Link'] = "https://www.amazon.in"+str(div.find\_all('a',{'class' : 'a-link-normal s-underline-text s-underline-link-text s-link-style'})[1]['href'])

            else:

                Fields['Review\_Link'] = "https://www.amazon.in"+str(div.find\_all('a',{'class' : 'a-link-normal s-underline-text s-underline-link-text s-link-style'})[0]['href'])

        if div.find\_all('span',{'class' : 'a-price-whole' }):

            Fields['Price'] = div.find\_all('span',{'class' : 'a-price-whole' })[0].get\_text()

        #print("\n"+str(Fields))

        url1 = Fields['Link']

        headers1 = {'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.9',

        'referer': f'https://www.amazon.in/s?k={product}',

        'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/100.0.4896.127 Safari/537.36'}

        response1 = requests.get(url = url1, headers = headers1)

        soup1 = BeautifulSoup(response1.content,'html.parser')

        Fields['Stock'] = soup1.find\_all('div',{'id' : 'availability'})[0].get\_text().strip().replace('.','')

        if soup1.find\_all('table',{'id' : 'productDetails\_techSpec\_section\_1'}):

            table\_tag = soup1.find\_all('table',{'id' : 'productDetails\_techSpec\_section\_1'})[0]

            tr\_tag = table\_tag.find\_all('tr')

            for tr in tr\_tag:

                Fields[tr.find\_all('th')[0].get\_text().strip().replace(' ','')] = tr.find\_all('td')[0].get\_text().strip().replace('\u200e','')

        else:

            table\_tag = soup1.find\_all('table',{'class' : 'a-normal a-spacing-micro'})[0]

            tr\_tag = table\_tag.find\_all('tr')

            for tr in tr\_tag:

                Fields[tr.find\_all('td',{'class':'a-span3'})[0].get\_text().strip().replace(' ','')] = tr.find\_all('td',{'class':'a-span9'})[0].get\_text().strip().replace('\u200e','')

        #print(str(Fields))

        df1 = df1.append(Fields, ignore\_index=True)

df2 = df1.sort\_values("Price")[1:100]

with pd.ExcelWriter(product+'\_amazon.xlsx') as writer:

    # Storing Total Product details in Excel, Sheet Name : Product\_Detial

    df1.to\_excel(writer, sheet\_name='Product\_Detial')

    # Storing Total Product details in Excel, Sheet Name : Mimimum Price

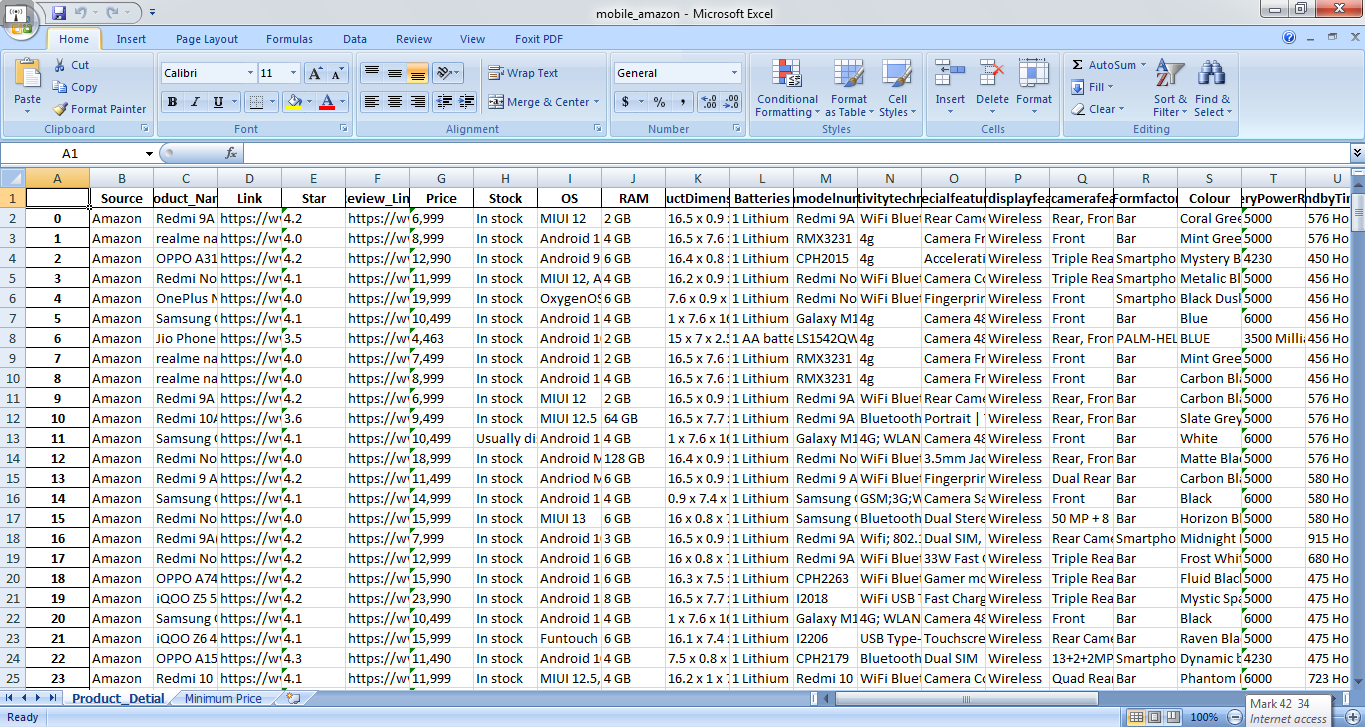
    df2.to\_excel(writer, sheet\_name='Minimum Price')

print('Done')

**Output:** click the below icon to view Excel Workbook



Sample Screenshot:



**Output file contains two sheets:**

Product\_Detail: Contains full details of products which we have given as Input.

Minimum\_Price: Contains minimum price list of given products.