**WEBSCRAPPING**

pip install requests :-- in cmd to have connection with outside world eg: http world i.e internet

pip install beautifulsoup4 :-- in cmd || it is python library || used to extract data from html page

PARSERS :-- to parse the markup language

Data is in markup language and markup can be 2 types :-- Html hyper text / xml i.e extended markup language

lxml

html.parser

html5lib

pip install lxml :-- to install the lxml parser

**WEBSCRAPPING THE “LAPTOP” SEARCH FROM FLIPKART WEBSITE** :---

from bs4 import BeautifulSoup

import requests

import csv

class santosh:

    url = "https://www.flipkart.com/search?q=laptop&otracker=search&otracker1=search&marketplace=FLIPKART&as-show=on&as=off"

    data = requests.get(url) #request will establish the connection with internet

    soup = BeautifulSoup(data.content, 'lxml')

    def flipkart(self):

        print(self.soup.prettify())

In the above program prettify() is used to display the entire html code in a nice manner in the cmd

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Now, from flipkart website we will print the 1st item under laptop search

def name\_flipkart(self):

        name = self.soup.find('div', class\_='\_4rR01T')

        return name

<div class="\_4rR01T">Lenovo Thinkpad Ryzen 5 Hexa Core 5600U - (8 GB/512 GB SSD/DOS) E14 AMD Gen3 Thin and Light Laptop</div>

So it wll print that entire div class

But to avoid the div class and to print only the product then we should use

    def name\_flipkart(self):

        name = self.soup.find('div', class\_='\_4rR01T')

        return name.text

Lenovo Thinkpad Ryzen 5 Hexa Core 5600U - (8 GB/512 GB SSD/DOS) E14 AMD Gen3 Thin and Light Laptop

By this , it will return only the text which is present in the div class

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Similarly to get the price with the item name then,

    def price\_flipkart(self):

        price = self.soup.find('div', class\_='\_30jeq3 \_1\_WHN1')

        return price.text

s = santosh()

print(s.name\_flipkart())

print(s.price\_flipkart())

ASUS Ryzen 7 Octa Core 5800HS - (16 GB/512 GB SSD/Windows 10 Home/4 GB Graphics/NVIDIA GeForce GTX 165...

₹79,990

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

From the above we can display the NAME and PRICE of an item under LAPTOP search from FLIPKART

Now using the scarping , we will display the entire list present for LAPTOP search from FLIPKART website

def all\_data(self): # method creation

        product = [] # taken a blank list

        prices = [] # taken a blank list

        for data in self.soup.findAll('div',class\_='\_3pLy-c row'):

            #for loop is used to search the entire page for the names and prices for the div class

            #findAll is that it will search everything on the page using the div class

            names = data.find('div', class\_='\_4rR01T') # generalized div class fro name

            prices = data.find('div', class\_='\_30jeq3 \_1\_WHN1') # generalized div class for price

            product.append(names.text) #here we are appending the list for product names

            prices.append(prices.text) #here we are appending the list for price

        print(product) # printing the product list

        print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

        print(prices) # printing the price list

s = santosh()

s.all\_data()

OUTPUT:---

['HP HP Envy Intel EVO Core i7 11th Gen - (16 GB/1 TB SSD/Windows 10 Home/2 GB Graphics) 13-ba1505TX Thi...', 'ASUS ROG Strix G15 (2021) Ryzen 7 Octa Core 4800H - (8 GB/512 GB SSD/Windows 10 Home/4 GB Graphics/NVI...', 'ASUS VivoBook 15 Core i3 10th Gen - (8 GB/1 TB HDD/Windows 11 Home) X515JA-BQ302W Thin and Light Lapto...',

'Lenovo IdeaPad 3 Core i3 10th Gen - (8 GB/256 GB SSD/Windows 11 Home) 15IML05 Thin and Light Laptop', 'ASUS VivoBook 15 (2022) Core i3 10th Gen - (8 GB/512 GB SSD/Windows 11 Home) X515JA-EJ362WS Thin and L...', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15ITL6 Thin and Light Laptop', 'ASUS VivoBook 15 (2022) Core i3 11th Gen - (8 GB/1 TB HDD/256 GB SSD/Windows 11 Home) X515EA-EJ332WS T...', 'ASUS ROG Zephyrus G14 Ryzen 7 Octa Core 4800HS - (8 GB/1 TB SSD/Windows 10 Home/4 GB Graphics/NVIDIA G...', 'DELL Inspiron Athlon Dual Core 3050U - (8 GB/256 GB SSD/Windows 11 Home) Inspiron 3525 Notebook', 'HP Ryzen 3 Dual Core 3250U - (8 GB/256 GB SSD/Windows 10 Home) 15s-GY0501AU Thin and Light Laptop', 'ASUS VivoBook 15 (2022) Celeron Quad Core - (4 GB/256 GB SSD/Windows 11 Home) X515MA-BR011W Thin and L...', 'ASUS ZenBook Duo 14 (2021) Touch Panel Intel EVO Core i7 11th Gen - (16 GB/1 TB SSD/Windows 10 Home) U...', 'HP Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15s-fq2629TU Thin and Light Laptop', 'MSI Modern 14 Ryzen 5 Hexa Core 4500U - (8 GB/512 GB SSD/Windows 10 Home) Modern 14 B4MW-423IN Thin an...', 'Infinix INBook X1 Core i3 10th Gen - (8 GB/256 GB SSD/Windows 11 Home) XL11 Thin and Light Laptop', 'ASUS Ryzen 9 Octa Core 5900HX - (16 GB/1 TB SSD/Windows 11 Home/4 GB Graphics) M7400QC-KM053WS Thin an...', 'HP Core i3 11th Gen - (8 GB/256 GB SSD/Windows 11 Home) 14s- dy2506TU Thin and Light Laptop', 'Lenovo Core i3 10PS G:\WORKSPACE\TESTING> & C:/Users/User/AppData/Local/Programs/Python/Python310/python.exe g:/WORKSPACE/TESTING/practise/app.py

['HP HP Envy Intel EVO Core i7 11th Gen - (16 GB/1 TB SSD/Windows 10 Home/2 GB Graphics) 13-ba1505TX Thi...', 'ASUS ROG Strix G15 (2021) Ryzen 7 Octa Core 4800H - (8 GB/512 GB SSD/Windows 10 Home/4 GB Graphics/NVI...', 'ASUS VivoBook 15 Core i3 10th Gen - (8 GB/1 TB HDD/Windows 11 Home) X515JA-BQ302W Thin and Light Lapto...',

'ASUS VivoBook 15 (2022) Core i3 10th Gen - (8 GB/512 GB SSD/Windows 11 Home) X515JA-EJ362WS Thin and L...', 'Lenovo IdeaPad 3 Core i3 10th Gen - (8 GB/256 GB SSD/Windows 11 Home) 15IML05 Thin and Light Laptop', 'DELL Inspiron Athlon Dual Core 3050U - (8 GB/256 GB SSD/Windows 11 Home) Inspiron 3525 Notebook', 'Lenovo IdeaPad 3 Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15ITL6 Thin and Light Laptop', 'ASUS ROG Zephyrus G14 Ryzen 7 Octa Core 4800HS - (8 GB/1 TB SSD/Windows 10 Home/4 GB Graphics/NVIDIA G...', 'ASUS VivoBook 15 (2022) Core i3 11th Gen - (8 GB/1 TB HDD/256 GB SSD/Windows 11 Home) X515EA-EJ332WS T...', 'HP Core i3 11th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15s-fq2629TU Thin and Light Laptop', 'ASUS VivoBook 15 (2022) Celeron Quad Core - (4 GB/256 GB SSD/Windows 11 Home) X515MA-BR011W Thin and L...', 'ASUS ZenBook Duo 14 (2021) Touch Panel Intel EVO Core i7 11th Gen - (16 GB/1 TB SSD/Windows 10 Home) U...', 'Infinix INBook X1 Core i3 10th Gen - (8

GB/256 GB SSD/Windows 11 Home) XL11 Thin and Light Laptop', 'MSI Modern 14 Ryzen 5 Hexa Core 4500U - (8 GB/512 GB SSD/Windows 10 Home) Modern 14 B4MW-423IN Thin an...', 'HP Ryzen 3 Dual Core 3250U - (8 GB/256 GB SSD/Windows 10 Home) 15s-GY0501AU Thin and Light Laptop', 'ASUS Ryzen 9 Octa Core 5900HX - (16 GB/1 TB SSD/Windows 11 Home/4 GB Graphics) M7400QC-KM053WS Thin an...', 'HP Ryzen 5 Hexa Core 5500U - (8 GB/512 GB SSD/Windows 11 Home) 14s-fq1092au Thin and Light Laptop', 'HP Core i3

11th Gen - (8 GB/256 GB SSD/Windows 11 Home) 14s- dy2506TU Thin and Light Laptop', 'Lenovo Core i3 10th Gen - (8 GB/512 GB SSD/Windows 11 Home) 15IML05 Thin and Light Laptop', 'MSI Stealth 15M Core i7 11th Gen - (16 GB/1 TB SSD/Windows 10 Home/6 GB Graphics/NVIDIA GeForce RTX 30...', 'DELL Vostro Core i3 10th Gen - (8 GB/1 TB

HDD/256 GB SSD/Windows 11 Home) Vostro 3510, X9K0T Thin and ...', 'ASUS Ryzen 3 Dual Core 3250U 3rd Gen - (8 GB/256 GB SSD/Windows 10 Home) M515DA-BQ322TS Thin and

Light...', 'Lenovo IdeaPad 3 Dual Core 3020e - (4 GB/256 GB SSD/Windows 11 Home) 14ADA05 Thin and Light Laptop', 'ASUS ROG Strix G15 Ryzen 9 Octa Core 5900HX - (16

GB/512 GB SSD/Windows 10 Home/4 GB Graphics/NVIDIA G...']

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<div class="\_30jeq3 \_1\_WHN1">₹95,990₹95,990</div>

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Now we can do web scrapping using selenium also as shown in below, i.e we created a program to scrape the followers count from instagram account

from selenium import webdriver

from selenium.webdriver.common.by import By

import requests

import time

class santosh:

    url = ""

    soup = requests.get(url)

    driver = webdriver.Firefox()

    def instagram\_login(self):

        username = "ABC"

        password = "ABC"

        self.driver.get(self.url)

        time.sleep(3)

        username1 = self.driver.find\_element(by=By.XPATH,value='COPY FULL XPATH')

        password1 = self.driver.find\_element(by=By.XPATH,value='COPY FULL XPATH')

        submit\_button = self.driver.find\_element(by=By.XPATH,value='COPY FULL XPATH')

        username1.send\_keys(username)

        password1.send\_keys(password)

        submit\_button.click()

        time.sleep(3)

        not\_now = self.driver.find\_element(by=By.XPATH,value='COPY FULL XPATH')

        not\_now.click()

    def instagram\_followers(self):

        self.instagram\_login()

        time.sleep(3)

        followers\_xpath = 'COPY FOLLOWERS COUNT XPATH'

        followers = self.driver.find\_element(by=By.XPATH,value=followers\_xpath)

        print(followers.text)

s = santosh()

s.instagram\_followers()