WEB SCRAPING

2.###Scrape the heading, date, content and the likes for the video from the link for the youtube video from the post.

```
!pip install beautifulsoup4

!pip install requests

import bs4

print(bs4.__version__)

import requests

from bs4 import BeautifulSoup

import pandas as pd

url = 'https://www.patreon.com/coreyms'

response = requests.get(url)

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


posts = []

for post in soup.find_all('div', class_='post'):

    heading = post.find('h2').text

    date = post.find('time')['datetime']

    content = post.find('div', class_='content').text

    likes = post.find('span', class_='likes').text

    youtube_link = post.find('a', href=True)['href']
```

```
    posts.append({
        'Heading': heading,
        'Date': date,
        'Content': content,
        'Likes': likes,
        'YouTube Link': youtube_link
    })

df = pd.DataFrame(posts)
df.to_csv('patreon_posts.csv', index=False)
```

3.######Define the Function to Scrape Data

```
def scrape_nobroker(locality):
    url = f'https://www.nobroker.in/property/sale/{locality}_bangalore'
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')

    houses = []
    for house in soup.find_all('div', class_='card'):
        title = house.find('h2', class_='heading-6').text.strip()
        location = house.find('div', class_='project-location').text.strip()
        area = house.find('div', class_='project-area').text.strip()
        price = house.find('div', class_='project-price').text.strip()
        emi = house.find('div', class_='project-emi').text.strip()
```

```
        houses.append({
            'Title': title,
            'Location': location,
            'Area': area,
            'Price': price,
            'EMI': emi
        })

    return houses
####Scrape Data for Specified Localities:
localities = ['indira-nagar', 'jayanagar', 'rajaji-nagar']
all_houses = []

for locality in localities:
    houses = scrape_nobroker(locality)
    all_houses.extend(houses)

df = pd.DataFrame(all_houses)
df.to_csv('nobroker_houses.csv', index=False)
4.###Scrape first 10 product details from Bewakoof
import requests
from bs4 import BeautifulSoup
```

```
url = "https://www.bewakoof.com/bestseller?sort=popular"

response = requests.get(url)

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


products = soup.find_all('div', class_='productCardBox', limit=10)


for product in products:

    name = product.find('h3').text

    price = product.find('span', class_='discountedPriceText').text

    image_url = product.find('img')['src']

    print(f"Product Name: {name}")

    print(f"Price: {price}")

    print(f"Image URL: {image_url}")

    print("-" * 20)
```

5.###Scrape headings, date, and news link from CNBC

```
import requests

from bs4 import BeautifulSoup


url = "https://www.cnbc.com/world/?region=world"

response = requests.get(url)

soup = BeautifulSoup(response.text, 'html.parser')
```

```
articles = soup.find_all('div', class_='Card-titleContainer', limit=10)


for article in articles:
    heading = article.find('a').text
    date = article.find('time')['datetime'] if article.find('time') else 'No date available'
    news_link = article.find('a')['href']
    print(f"Heading: {heading}")
    print(f"Date: {date}")
    print(f"News Link: {news_link}")
    print("-" * 20)
```

6.####Scrape paper title, date, and author from KeAi Publishing

```
import requests
from bs4 import BeautifulSoup

url = "https://www.keaipublishing.com/en/journals/artificial-intelligence-in-agriculture/most-downloaded-articles/"
response = requests.get(url)
soup = BeautifulSoup(response.text, 'html.parser')


papers = soup.find_all('div', class_='article-content', limit=10)


for paper in papers:
    title = paper.find('h2').text
    date = paper.find('span', class_='date').text if paper.find('span', class_='date') else 'No
```

```
date available'

    author = paper.find('span', class_='authors').text if paper.find('span', class_='authors')
else 'No author available'

    print(f"Paper Title: {title}")

    print(f"Date: {date}")

    print(f"Author: {author}")

    print("-" * 20)
```