

1. Write a python program to display IMDB's Top rated 100 Indian movies' data <https://www.imdb.com/list/ls056092300/> (<https://www.imdb.com/list/ls056092300/>) (i.e. name, rating, year of release) and make data frame

```
In [2]: from bs4 import BeautifulSoup
import requests
```

```
In [2]: url="https://www.imdb.com/list/ls056092300/"
page=requests.get(url)
page
```

Out[2]: <Response [403]>

2. Write a python program to scrape details of all the posts from <https://www.patreon.com/coreyms> (<https://www.patreon.com/coreyms>). Scrape the heading, date, content and the likes for the video from the link for the youtube video from the post.

```
In [3]: url="https://www.patreon.com/coreyms"
page=requests.get(url)
page
```

Out[3]: <Response [200]>

```
In [41]: soup=BeautifulSoup(page.content, "html.parser")
soup
```

```
Out[41]: <!DOCTYPE html>
<html><head><meta charset="utf-8"/><script type="text/javascript">
    window.patreonConfig = {"apiServer":"www.patreon.com/ap
i","nionDisableDefaultIncludes":true,"formattingLocale":"en-US","event
LoggingPlatform":"Web","isWebView":false,"pathname":"/\u003cstring:cam
paign_identifier\u003e","is_impersonating":false,"csrfSignature":"xL4K
oi_n1HaGIjyA2z_gYgc3ZnLjY38PQjbt-DXA5A"};
</script><meta content="width=device-width, initial-scale=1.
0, viewport-fit=cover" name="viewport"/><meta content="artists, creato
rs, patron, sponsor, music, videos" name="keywords"/><meta content="20
1564" property="fb:admins"/><meta content="130127590512253" property
="fb:app_id"/><meta content="IE=edge" http-equiv="X-UA-Compatible"/><m
eta content="@Patreon" property="twitter:site"/><meta content="Patreo
n" property="og:site-name"/><meta content="Patreon" prefix="og: htt
p://ogp.me/ns#" (http://ogp.me/ns#" property="og:site_name"/><meta co
ntent="Patreon" name="apple-mobile-web-app-title"/><meta content="Patr
eon" name="application-name"/><link as="fetch" href="https://www.patre
on.com/api/experiments/assignments?json-api-version=1.0&json-api-u
se-default-includes=false&include=[]" rel="prefetch"/><link href
"..."
```

```
In [42]: content=soup.find('div',class_="sc-bdvvtL lhrfPG")
content
```

```
Out[42]: <div class="sc-bdvvtL lhrfPG"><div>I appreciate any support! Anyone who i
s a contributor through Patreon will have the option to be added to my Co
ntributors page on my personal website (coreyms.com) <br/></div></div>
```

```
In [43]: content.text
```

```
Out[43]: 'I appreciate any support! Anyone who is a contributor through Patreon wi
ll have the option to be added to my Contributors page on my personal web
site (coreyms.com)\xa0'
```

```
In [51]: heading=[]
for i in soup.find_all('h1',class_="style-scope ytd-watch-metadata"):
    heading.append(i.text)
```

```
In [54]: heading
```

```
Out[54]: []
```

```
In [94]: heading=soup.find('span',class_="sc-1cvoily-0 eGFfXM")
heading
```

```
In [95]: heading.text
```

```
-----
--
AttributeError                                Traceback (most recent call las
t)
Cell In[95], line 1
----> 1 heading.text

AttributeError: 'NoneType' object has no attribute 'text'
```

```
In [97]: content=soup.find('div',class_="sc-cfnzm4-0 daxSFj")
content
```

3. Write a python program to scrape house details from mentioned URL. It should include house title, location, area, EMI and price from <https://www.nobroker.in/> (<https://www.nobroker.in/>). Enter three localities which are Indira Nagar, Jayanagar, Rajaji Nagar.

```
In [65]: import pandas as pd
import requests
from bs4 import BeautifulSoup
```

```
In [66]: url="https://www.nobroker.in/"
page=requests.get(url)
page
```

Out[66]: <Response [200]>

```
In [67]: soup=BeautifulSoup(page.content)
soup
```

```
Out[67]: <!DOCTYPE html>
<html lang="en"><head>
<meta content="794951570520699" property="fb:pages"/>
<link href="https://www.nobroker.in" rel="canonical"/>
<link href="//www.googletagmanager.com" rel="dns-prefetch preconnect"/>
<link href="//www.google-analytics.com" rel="dns-prefetch preconnect"/>
<link href="//assets.nobroker.in" rel="dns-prefetch"/>
<link href="//images.nobroker.in" rel="dns-prefetch"/>
<link href="https://assets.adobedtm.com" rel="dns-prefetch preconnect"/>
<link href="//assets.nobroker.in/static/img/favicon.png" id="favicon"
rel="shortcut icon"/>
<link href="https://images.nobroker.in/static/img/fav64.png" rel="apple-touch-icon"/>
<link href="https://hs.nobroker.in" rel="dns-prefetch preconnect"/>
<!--<meta charset="utf-8"/>-->
<meta charset="utf-8"/>
<meta content="app_id.com.nobroker.app?referrer=utm_source%3Dnobro
```

4. Write a python program to scrape first 10 product details which include product name , price , Image URL from <https://www.bewakoof.com/bestseller?sort=popular> (<https://www.bewakoof.com/bestseller?sort=popular>)

```
In [18]: url="https://www.bewakoof.com/bestseller?sort=popular"
```

```
In [19]: page=requests.get(url)
page
```

Out[19]: <Response [200]>

```
In [20]: soup=BeautifulSoup(page.content)
        soup
```

```
Out[20]: <!DOCTYPE html>
<html><head>
<meta content="4d8KRpUCzf-ghquF_HLNphibw_xRvUS3kAvzpnF2K1A" name="google-site-verification"/>
<meta content="-TTsyPFCOB4bQlMHawi04oZGcGX9pxqenB_NZNhhhgA" name="google-site-verification"/>
<meta content="wQa0CQvt7fd2VIOa9qhkevhwuBvhRwiaB3hVizI90Y" name="google-site-verification"/>
<meta content="o7KS3-fVt15m4Cc1HeYh5HIpv532A4FvvYQzIf-LnsQ" name="google-site-verification"/>
<link href="https://images.bewakoof.com" rel="preconnect"/>
<link href="https://images.bewakoof.com" rel="dns-prefetch"/>
<link href="https://static.bewakoof.com" rel="preconnect"/>
<link href="https://static.bewakoof.com" rel="dns-prefetch"/>
<link href="https://cdn.polyfill.io" rel="preconnect"/>
<link href="https://cdn.polyfill.io" rel="dns-prefetch"/>
<link href="https://www.googletagmanager.com" rel="preconnect"/>
<link href="https://www.googletagmanager.com" rel="dns-prefetch"/>
<link href="https://www.google-analytics.com" rel="dns-prefetch"/>
```

```
In [27]: product= soup.find('div',class_="productNaming bkf-ellipsis")
        product
```

```
Out[27]: <div class="productNaming bkf-ellipsis"><h3 class="brand-name undefined">
Bewakoof®</h3><h2 class="clr-shade4 h3-p-name undefined false">Women's Red
Bonjour Paris Graphic Printed Plus Size Boyfriend T-shirt</h2></div>
```

```
In [28]: product.text
```

```
Out[28]: "Bewakoof®Women's Red Bonjour Paris Graphic Printed Plus Size Boyfriend T
-shirt"
```

```
In [42]: product=[]
        for i in soup.find_all('div',class_="productNaming bkf-ellipsis"):
            product.append(i.text)
```

```
In [43]: product
```

```
Out[43]: ["Bewakoof®Women's Red Bonjour Paris Graphic Printed Plus Size Boyfriend
T-shirt",
"Bewakoof Air® 1.0Men's Black Oversized Plus Size Cargo Joggers",
"Bewakoof Air® 1.0Men's Black Oversized Cargo Joggers",
"Bewakoof®Men's Black Plus Size Casual Joggers",
"Bewakoof®Men's Blue Plus Size Casual Joggers",
"Bewakoof®Men's Black Oversized Cargo Pants",
"Bewakoof®Women's Black Minimal Believe Graphic Printed Plus Size Boyfri
end T-shirt",
"bewakoof x marvelMen's Grey We Are Venom Graphic Printed Oversized Ves
t",
"Bewakoof Air® 1.0Women's Black Oversized Cargo Joggers",
"Bewakoof®Men's Grey Oversized Cargo Pants"]
```

```
In [40]: price=soup.find("span",class_="sellingPrice mr-1")
price
```

```
In [38]: price.text
```

```
-----
--
AttributeError                                Traceback (most recent call las
t)
Cell In[38], line 1
----> 1 price.text

AttributeError: 'NoneType' object has no attribute 'text'
```

```
In [33]: images=[]
for i in soup.find_all('img',class_="swiper-lazy swiper-lazy-loaded"):
    images.append(i['data-src'])
images
```

```
Out[33]: []
```

5. Please visit <https://www.cnbc.com/world/?region=world> (<https://www.cnbc.com/world/?region=world>) and scrap- a) headings b) date c) News link

```
In [58]: page=requests.get("https://www.cnbc.com/world/?region=world")
page
```

```
Out[58]: <Response [200]>
```

```
In [59]: soup=BeautifulSoup(page.content)
soup
```

```
Out[59]: <!DOCTYPE html>
<html itemscope="" itemtype="https://schema.org/WebPage" lang="en" pre
fix="og=https://ogp.me/ns#"><head><meta content="max-image-preview:lar
ge" name="robots"/><meta content="telephone=no" name="format-detectio
n"/><link href="https://sc.cnbcfm.com/applications/cnbc.com/staticcont
ent/img/favicon.ico" rel="icon" type="image/x-icon"/><style type="tex
t/css">@charset "UTF-8";.Modal-modalBackground{background:#000000b3;he
ight:100%;left:0;overflow-y:auto;position:fixed;top:0;transition:backg
round-color .4s;width:100%;z-index:100001}.Modal-bottomModal.Modal-mod
al{background:#f8f8f8;border-radius:3px;bottom:0;box-shadow:5px 5px 20
px #1717171a;display:inline-block;height:528px;left:0;margin-top:0!imp
ortant;max-width:100%;position:fixed;top:auto;transform:none;width:10
0%}@media (max-width:1019px){.Modal-bottomModal.Modal-modal{height:642
px}}@media (max-width:759px){.Modal-bottomModal.Modal-modal{height:10
0%;position:relative;top:0}}.Modal-modal{background-color:#fff;border-
radius:3px;box-shadow:5px 5px 20px #1717171a;display:inline-block;lef
t:50%;margin-top:10vh;max-width:100%;overflow:auto;position:relative;t
ransform:translateX(-50%)}@media (max-width:759px){.Modal-modal{heigh
t:100%;left:auto;margin:0;transform:none;width:100%}}.Modal-modalConte
nt{overflow:auto}@media (max-width:759px){.Modal-modalContent{height
```

```
In [60]: heading=soup.find('span',class_="nav-menu-buttonText")
         heading
```

```
Out[60]: <span class="nav-menu-buttonText">Markets</span>
```

```
In [61]: heading.text
```

```
Out[61]: 'Markets'
```

```
In [62]: date=soup.find('time',class_="LatestNews-timestamp")
         date
```

```
Out[62]: <time class="LatestNews-timestamp">20 Hours Ago</time>
```

```
In [63]: date.text
```

Out[63]: '20 Hours Ago'

```
In [64]: link=soup.find('div',class_="MostPopular-linkWrap")
link
```

In []:

6. Please visit <https://www.keaipublishing.com/en/journals/artificial-intelligence-in-agriculture/most-downloaded%02articles/> and scrap- a) Paper title b) date c) Author

In [69]: `page=requests.get("https://www.keaipublishing.com/en/journals/artificial-in`

In [70]: page

Out[70]: <Response [400]>

In []: