

In [1]:

Answer1

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

In [2]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

In [3]:

```
driver.get("https://www.naukri.com/")
```

In [4]:

```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input")
designation.send_keys('Data Analyst')
```

In [5]:

```
location=driver.find_element(By.XPATH,"/html/body/div[1]/div[7]/div/div/div
[5]/div/div/div/div[1]/div/input")
location.send_keys('Banglore')
```

In [6]:

```
search=driver.find_element(By.CLASS_NAME,"qsbSubmit")
search.click()
```

In [7]:

```
job_title=[]
job_location=[]
company_name=[]
experience_required=[]
```

In [10]:

```
title_tags=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"]')
for i in title_tags[0:10]:
    title=i.text
    job_title.append(title)

location_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis fleft
locWdth"]')
for i in location_tags[0:10]:
    location=i.text
    job_location.append(location)

company_tags=driver.find_elements(By.XPATH,'//a[@class="subTitle ellipsis
fleft"]')
for i in company_tags[0:10]:
    company=i.text
    company_name.append(company)

experience_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis
fleft expwdth"]')
```

```
for i in experience_tags[0:10]:
    experience=i.text
    experience_required.append(experience)
```

In [12]:

```
print(len(job_title),len(job_location),len(company_name),len(experience_required))
20 20 20 20
```

In [22]:

```
import pandas as pd
df=pd.DataFrame({"Title":job_title,"Location":job_location,"Company":company_name,"Experience":experience_required})
df.head(10)
```

Out[22]:

	Title	Location	Company	Experience
0	Data Analyst	Bangalore/Bengaluru	Cynosure Corporate Solutions	2-7 Yrs
1	Data Analyst III	Bangalore/Bengaluru	Walmart	3-7 Yrs
2	Data Analyst III	Bangalore/Bengaluru	Walmart	3-7 Yrs
3	Compliance Rule Writer/Data Analyst	Bangalore/Bengaluru	Paypal	5-8 Yrs
4	Data Analyst	Bangalore/Bengaluru	Ingersoll Rand	3-6 Yrs
5	Data Analyst - EdTech	Bangalore/Bengaluru	Talentstack	2-6 Yrs
6	Data Analyst	Bangalore/Bengaluru	Unusual Hire	1-4 Yrs
7	Data Engineer/Data Analyst	Hybrid - Kolkata, Hyderabad/Secunderabad, Pune...	Tech Mahindra	6-11 Yrs
8	Data Analyst - Contractual	Bangalore/Bengaluru	Search Advisers Services Guj	2-3 Yrs
9	Data Analyst - Contractual	Bangalore/Bengaluru	Search Advisers Services Guj	2-3 Yrs

In []:

Answer2

In [5]:

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

In [6]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32(1)\chromedriver.exe")
```

In [7]:

```
driver.get("https://www.naukri.com/")
```

In [8]:

```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input")
designation.send_keys('Data Scientist')
```

```
location=driver.find_element(By.XPATH, '/html/body/div[1]/div[7]/div/div/div[5]/div/div/div/div[1]/div/input')
location.send_keys("Bangalore")
```

In [9]:

```
search=driver.find_element(By.CLASS_NAME, "qsbSubmit")
search.click()
```

In [10]:

```
jobtitle=[]
joblocation=[]
jobname=[]
```

In [11]:

```
title_tags=driver.find_elements(By.XPATH, '//a[@class="title ellipsis"]')
for i in title_tags[0:10]:
    title=i.text
    jobtitle.append(title)
```

In [12]:

```
location_tags=driver.find_elements(By.XPATH, '//span[@class="ellipsis fleft locWdth"]')
for i in location_tags[0:10]:
    location=i.text
    joblocation.append(location)
```

```
Name_tags=driver.find_elements(By.XPATH, '//a[@class="subTitle ellipsis fleft"]')
for i in Name_tags[0:10]:
    tags=i.text
    jobname.append(tags)
```

In [13]:

```
print(len(jobtitle),len(joblocation),len(jobname))
10 10 10
```

In [14]:

```
import pandas as pd
db=pd.DataFrame({"Profile":jobtitle,"Location":location,"Company Name":jobname})
db
```

Out[14]:

	Profile	Location	Company Name
0	Data Science Professional - IBM SPSS Statistic...	Bangalore/Bengaluru, Mumbai	Hexaware Technologies
1	Data Science Specialist	Bangalore/Bengaluru, Mumbai	Accenture
2	Analystics & Modeling Specialist	Bangalore/Bengaluru, Mumbai	Accenture
3	Senior data scientist	Bangalore/Bengaluru, Mumbai	Fractal Analytics
4	Data Scientist_NLP	Bangalore/Bengaluru, Mumbai	Fractal Analytics

	Profile	Location	Company Name
5	Data Scientist	Bangalore/Bengaluru, Mumbai	Fractal Analytics
6	Machine Learning (AI) Architect	Bangalore/Bengaluru, Mumbai	Persistent
7	Manager - Innovations Hub - Machine Learning	Bangalore/Bengaluru, Mumbai	PwC
8	Data Scientist	Bangalore/Bengaluru, Mumbai	Cognizant
9	Staff Data Scientist	Bangalore/Bengaluru, Mumbai	Baker Hughes

In []:

Answer 3

In [15]:

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

In [16]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32(1)\chromedriver.exe")
```

In [17]:

```
driver.get("https://www.naukri.com/")
```

In [18]:

```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input")
designation.send_keys('Data Scientist')
```

In [19]:

```
search=driver.find_element(By.CLASS_NAME,"qsbSubmit")
search.click()
```

In [24]:

```
location=driver.find_element(By.XPATH,"/html/body/div[1]/div[4]/div/div/section[1]/div[2]/div[5]/div[2]/div[2]/label/p/span[1]")
location.click()
```

In [25]:

```
salary=driver.find_element(By.XPATH,"/html/body/div[1]/div[4]/div/div/section[1]/div[2]/div[3]/div[2]/div[2]/label/p/span[1]")
salary.click()
```

In [26]:

```
Jobtitle=[]
Joblocation=[]
Companyname=[]
Experiencerequired=[]
```

In [27]:

```
title_tags=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"]')
for i in title_tags[0:10]:
    title=i.text
    Jobtitle.append(title)
```

```
location_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis fleft locWdth"]')
for i in location_tags[0:10]:
    locaton=i.text
    Joblocation.append(location)
```

```
companyname_tags=driver.find_elements(By.XPATH,'//a[@class="subTitle ellipsis fleft"]')
for i in companyname_tags[0:10]:
    name=i.text
    Companyname.append(name)
```

```
Experience_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis fleft expwidth"]')
for i in Experience_tags[0:10]:
    Experience=i.text
    Experiencerequired.append(Experience)
```

In [28]:

```
print(len(Jobtitle),len(Joblocation),len(Companyname),len(Experiencerequired))
10 10 10 10
```

In [29]:

```
import pandas as pd
df=pd.DataFrame({'Profile':Jobtitle,'Location':Joblocation,'Name':Companyname,'Experience':Experiencerequired})
df
```

Out[29]:

	Profile	Location	Name	Experience
0	Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Cognizant	6-10 Yrs
1	Junior Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Analytos	0-2 Yrs
2	Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Blackbuck	3-7 Yrs
3	MLOps - Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Paytm	3-8 Yrs
4	Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Tabsquare	1-3 Yrs
5	Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Analytos	2-4 Yrs
6	Junior Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	Adidas	1-6 Yrs
7	Analyst-Data Science	<selenium.webdriver.remote.webelement.WebEleme...	AMERICAN EXPRESS	0-3 Yrs

	Profile	Location	Name	Experience
8	Analyst-Data Science	<selenium.webdriver.remote.webelement.WebEleme...	AMERICAN EXPRESS	0-3 Yrs
9	Senior Data Scientist	<selenium.webdriver.remote.webelement.WebEleme...	RecruitEForU	3-8 Yrs

In []:

answer 4

In [23]:

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

In [24]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

In [25]:

```
driver.get("https://www.flipkart.com/")
```

In [26]:

```
product=driver.find_element(By.CLASS_NAME,"_3704LK")
product.send_keys('Sunglasses')
```

In [28]:

```
search=driver.find_element(By.CLASS_NAME,"L0Z3Pu")
search.click()
```

In [29]:

```
brand=[]
description=[]
price=[]
discount=[]
```

In [30]:

```
start=0
end=3
for page in range(start,end):
    titletags=driver.find_elements(By.XPATH,'//div[@class="_2WkVRV"]')
    for i in titletags:
        brand.append(i.text)

start=0
end=3
for page in range(start,end):
    destags=driver.find_elements(By.XPATH,'//a[@class="IRpwTa"]')
    for i in destags:
        description.append(i.text)

start=0
end=3
```

```

for page in range(start,end):
    pricetags=driver.find_elements(By.XPATH,'//div[@class="_30jeq3"]')
    for i in pricetags:
        price.append(i.text)

start=0
end=3
for page in range(start,end):
    discounttags=driver.find_elements(By.XPATH,'//div[@class="_3Ay6Sb"]')
    for i in discounttags:
        discount.append(i.text)

```

In [31]:

```

print(len(brand),len(description),len(price),len(discount))
120 117 120 120

```

In [32]:

```

import pandas as pd
df=pd.DataFrame({"Brand":brand,"Product
description":description,"Price":price,"Discount":discount})
df

```

```

-----
ValueError                                Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_80732\2741611294.py in <module>
      1 import pandas as pd
----> 2 df=pd.DataFrame({"Brand":brand,"Product
description":description,"Price":price,"Discount":discount})
      3 df

~\anaconda3\lib\site-packages\pandas\core\frame.py in __init__(self, data,
index, columns, dtype, copy)
    634         elif isinstance(data, dict):
    635             # GH#38939 de facto copy defaults to False only in non-
dict cases
--> 636         mgr = dict_to_mgr(data, index, columns, dtype=dtype,
copy=copy, typ=manager)
    637         elif isinstance(data, ma.MaskedArray):
    638             import numpy.ma.mrecords as mrecords

~\anaconda3\lib\site-packages\pandas\core\internals\construction.py in
dict_to_mgr(data, index, columns, dtype, typ, copy)
    500         # TODO: can we get rid of the dt64tz special case above?
    501
--> 502         return arrays_to_mgr(arrays, columns, index, dtype=dtype,
typ=typ, consolidate=copy)
    503
    504

```

```

~\anaconda3\lib\site-packages\pandas\core\internals\construction.py in
arrays_to_mgr(arrays, columns, index, dtype, verify_integrity, typ,
consolidate)
    118         # figure out the index, if necessary
    119         if index is None:
--> 120             index = _extract_index(arrays)
    121         else:
    122             index = ensure_index(index)

~\anaconda3\lib\site-packages\pandas\core\internals\construction.py in
_extract_index(data)
    672         lengths = list(set(raw_lengths))
    673         if len(lengths) > 1:
--> 674             raise ValueError("All arrays must be of the same
length")
    675
    676         if have_dicts:

```

ValueError: All arrays must be of the same length

In [ ]:

Answer 5

In [33]:

```

import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time

```

In [34]:

```

driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")

```

In [36]:

```

driver.get("https://www.flipkart.com/apple-iphone-11-black-64-gb/product-
reviews/itm4e5041ba101fd?pid=MOBFWQ6BXGJCEYNY&lid=LSTMOBFWQ6BXGJCEYNYZXSHRJ
&market")

```

In [37]:

```

Rating=[]
Summary=[]
Fullreview=[]

```

In [38]:

```

start=0
end=10
for page in range(start,end):
    ratingtags=driver.find_elements(By.XPATH, '//div[@class="_3LWZlK
_1BLPMq"]')
    for i in ratingtags:
        Rating.append(i.text)

start=0

```



```

end=10
for page in range(start,end):
    reviewssummary=driver.find_elements(By.XPATH,'//p[@class="_2-N8zT"]')
    for i in reviewssummary:
        Summary.append(i.text)

start=0
end=10
for page in range(start,end):
    review=driver.find_elements(By.XPATH,'//div[@class="t-ZTKy"]')
    for i in review:
        Fullreview.append(i.text)

```

```

print(len(Rating),len(Summary),len(Fullreview))
100 100 100

```

In [39]:

```

import pandas as pd
df=pd.DataFrame({"Rating":Rating,"Review Summary":Summary,"Full
Review":Fullreview})
df

```

In [41]:

	Rating	Review Summary	Full Review
0	5	Simply awesome	Really satisfied with the Product I received....
1	5	Perfect product!	Amazing phone with great cameras and better ba...
2	5	Best in the market!	Great iPhone very snappy experience as apple k...
3	4	Value-for-money	I'm Really happy with the product\nDelivery wa...
4	5	Highly recommended	It's my first time to use iOS phone and I am l...
...	...	...	...
95	5	Worth every penny	Previously I was using one plus 3t it was a gr...
96	4	Pretty good	I was using Iphone 6s and also Oneplus 6t. Bot...
97	5	Perfect product!	Value for money\n5 star rating\nExcellent came...
98	5	Highly recommended	What a camera .....just awesome ..you can feel...
99	5	Great product	Amazing Powerful and Durable Gadget.\n\nI'm am...

Out[41]:

100 rows x 3 columns

Answer 6

In [ ]:

```

import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time

```

In [44]:

In [45]:

```

driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")

In [46]:
driver.get("https://www.flipkart.com/")

In [47]:
product=driver.find_element(By.CLASS_NAME,"_3704LK")
product.send_keys('sneakers')

In [48]:
search=driver.find_element(By.CLASS_NAME,"L0Z3Pu")
search.click()

In [49]:
brand=[]
description=[]
price=[]

In [52]:
start=0
end=3
for page in range(start,end):
    brandtags=driver.find_elements(By.XPATH,'//div[@class="_2WkVRV"]')
    for i in brandtags:
        brand.append(i.text)

start=0
end=3
for page in range(start,end):
    descriptiontags=driver.find_elements(By.XPATH,'//a[@class="IRpwTa"]')
    for i in descriptiontags:
        description.append(i.text)

start=0
end=3
for page in range(start,end):
    pricing=driver.find_elements(By.XPATH,'//div[@class="_30jeq3"]')
    for i in pricing:
        price.append(i.text)

In [53]:
print(len(brand),len(description),len(price))
120 120 120

In [55]:
import pandas as pd
dataframe=pd.DataFrame({"Brand":brand,"Praduct
description":description,"Price":price})
dataframe.head(100)

```

Out[55]:

	Brand	Praduct description	Price
0	SFR	2006 Trenddy Fashion Sporty Casuals Sneakers R...	₹299
1	Kraasa	Sneakers For Women	₹299
2	BIRDE	Combo Pack Of 2 Casual Shoes Sneakers For Men	₹499
3	AMICO	Sneakers For Men	₹387

	Brand	Praduct description	Price
4	Labbin	Sneakers For Men	₹379
...	...	...	...
95	BIRDE	Casual Shoes Sneakers For Women	₹199
96	asian	WATERPROOF-05cFULLWHITE Sneakers For Men	₹629
97	aadi	Lightweight,Comfort,Summer,Trendy,Walking,Outd...	₹399
98	Xtoon	Modern Trendy Sneakers boot Sneakers For Men	₹349
99	PUMA	Puma Smash Vulc Sneakers For Men	₹1,394

100 rows × 3 columns

Answer 8

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

```
driver.get("https://www.azquotes.com/")
```

```
search=driver.find_element(By.XPATH, '/html/body/div[1]/div[1]/div[1]/div/di
v[3]/ul/li[5]/a')
search.click()
```

```
Quotes=[]
Author=[]
Type=[]
```

```
start=0
end=10
for page in range(start,end):
    typetags=driver.find_elements(By.XPATH, '//div[@class="tags"]')
    for i in typetags:
        Type.append(i.text)

start=0
end=10
for page in range(start,end):
    authortags=driver.find_elements(By.XPATH, '//div[@class="author"]')
    for i in authortags:
        Author.append(i.text)

start=0
```

In [ ]:

In [13]:

In [14]:

In [15]:

In [20]:

In [21]:

In [23]:

```

end=10
for page in range(start,end):
    quotesfull=driver.find_elements(By.XPATH, '//a[@class="title"] ')
    for i in quotesfull:
        Quotes.append(i.text)

print(len(Quotes),len(Author),len(Type))
1000 1000 1000

```

In [26]:

```

import pandas as pd
df=pd.DataFrame({'Quotes':Quotes,'Author':Author,'Type':Type})
df

```

In [27]:

Out[27]:

	Quotes	Author	Type
0	The essence of strategy is choosing what not t...	Michael Porter	Essence, Deep Thought, Transcendentalism
1	One cannot and must not try to erase the past ...	Golda Meir	Inspiration, Past, Trying
2	Patriotism means to stand by the country. It d...	Theodore Roosevelt	Country, Peace, War
3	Death is something inevitable. When a man has ...	Nelson Mandela	Inspirational, Motivational, Death
4	You have to love a nation that celebrates its ...	Erma Bombeck	4th Of July, Food, Patriotic
...	...	...	...
995	When the going gets weird, the weird turn pro.	Hunter S. Thompson	Music, Sports, Hunting
996	When a train goes through a tunnel and it gets...	Corrie Ten Boom	Trust, Encouraging, Uplifting
997	If you think you are too small to make a diffe...	Dalai Lama	Inspirational, Funny, Change
998	God doesn't require us to succeed, he only req...	Mother Teresa	Success, God, Mother
999	Change your thoughts and you change your world.	Norman Vincent Peale	Inspirational, Motivational, Change

1000 rows × 3 columns

In [ ]:

Answer 9

In [60]:

```

import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time

```

In [61]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

In [62]:

```
driver.get("https://www.jagranjosh.com/")
```

In [63]:

```
gkoption=driver.find_element(By.XPATH, '/html/body/div/div[1]/div/div[1]/div
/div[5]/div/div[1]/header/div[3]/ul/li[9]/a')
gkoption.click()
```

In []:

In []:

In []:

In []:

In [29]:

In [31]:

In [32]:

In [33]:

In [34]:

```
Primeminister=[]
Borndead=[]
term=[]
remarks=[]
```

In [35]:

```
ministertags=driver.find_elements(By.XPATH, '/html/body/div[1]/div[2]/div/di
v[2]/div/div[1]/div/div/div/div[5]/span/div[2]/table/tbody/tr[4]/td[2]/p')
for i in ministertags:
    minister=i.text
    Primeminister.append(minister)
```

```
Primeminister
```

Out[35]:

```
['Lal Bahadur Shastri']
```

In []:

In []:

In []:

Answer 10

In [65]:

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
```

```
from selenium.webdriver.common.by import By
import time
```

In [66]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

In [67]:

```
driver.get("https://www.motor1.com/")
```

In [68]:

```
leftlist=driver.find_element(By.XPATH, '/html/body/div[3]/div[2]/div/div/div
[1]/div')
leftlist.click()
```

In [69]:

```
news=driver.find_element(By.XPATH, '/html/body/div[4]/div[1]/div[3]/ul/li[1]
')
news.click()
```

In [72]:

```
features=driver.find_element(By.XPATH, '/html/body/div[4]/div[1]/div[3]/ul/li
[5]')
features.click()
```

In [73]:

```
list=driver.find_element(By.XPATH, '/html/body/div[3]/div[8]/div/div/div/div
/div/a[1]')
list.click()
```

In [74]:

```
expensive=driver.find_element(By.XPATH, '/html/body/div[3]/div[8]/div[1]/div
[1]/div/div/div[8]/div/div[1]')
expensive.click()
```

In [75]:

```
Carname=[]
price=[]
```

In [76]:

```
cartags=driver.find_elements(By.XPATH, '//h3[@class="subheader"]')
for i in cartags:
    car=i.text
    Carname.append(car)
```

Carname

Out[76]:

```
['De Tomaso P72',
 'Ferrari LaFerrari',
 'Pagani Huayra',
 'McLaren Elva',
 'Czinger 21C',
 'Ferrari Monza',
 'Gordon Murray T.33',
 'Koenigsegg Gemera',
 'Zenvo TSR-S',
 'Hennessey Venom F5',
 'Bentley Bacalar',
 'Hispano Suiza Carmen Boulogne',
```

```

'Bentley Mulliner Batur',
'Deus Vayanne',
'SSC Tuatara',
'Lotus Evija',
'Aston Martin Vulcan',
'Delage D12',
'McLaren Speedtail',
'Rimac Nevera',
'Pagani Utopia',
'Pininfarina Battista',
'Ferrari FXX K Evo',
'Gordon Murray T.50',
'Lamborghini Countach',
'Mercedes-AMG Project One',
'Aston Martin Victor',
'Hennessey Venom F5 Roadster',
'Koenigsegg Jesko',
'Aston Martin Valkyrie',
'W Motors Lykan Hypersport',
'McLaren Solus',
'Pagani Huayra Roadster BC',
'Bugatti Chiron Pur Sport',
'Lamborghini Sian',
'Koenigsegg CC850',
'Bugatti Chiron Super Sport 300+',
'Lamborghini Veneno',
'Bugatti Bolide',
'Bugatti Mistral',
'Pagani Huayra Imola',
'Bugatti Divo',
'SP Automotive Chaos',
'Pagani Codalunga',
'Mercedes-Maybach Exelero',
'Bugatti Centodieci',
'Bugatti Chiron Profilée',
'Rolls-Royce Sweptail',
'Bugatti La Voiture Noire',
'Rolls-Royce Boat Tail*',
'Most Expensive Cars In The World']

```

In []:

```

pricetags=driver.find_elements(By.XPATH,'//*[@id="article_box"]/div[1]/div[2]/div[1]/p[4]')
for i in pricetags:
    price=(i.text)

```

price

In []:

In []:

## Answer 7

In [95]:

```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.webdriver.common.by import By
import time
```

In [96]:

```
driver=webdriver.Chrome(r"C:\Users\zarna\Downloads\chromedriver_win32
(1)\chromedriver.exe")
```

In [97]:

```
driver.get("https://www.amazon.in/")
```

In [98]:

```
laptopsearch=driver.find_element(By.XPATH, "/html/body/div[1]/header/div/div
[1]/div[2]/div/form/div[2]/div[1]/input")
laptopsearch.send_keys("laptop")
```

In [99]:

```
search=driver.find_element(By.XPATH, "/html/body/div[1]/header/div/div[1]/di
v[2]/div/form/div[3]/div/span/input")
search.click()
```

In [100]:

```
cpu=driver.find_element(By.XPATH, "/html/body/div[1]/div[2]/div[1]/div[2]/di
v/div[3]/span/div[1]/div/div/div[6]/ul[6]/li[10]/span/a/span")
cpu.click()
```

In [101]:

```
Title=[]
Rating=[]
Price=[]
```

In [102]:

```
title_tags=driver.find_elements(By.XPATH, '//span[@class="a-size-medium a-
color-base a-text-normal"]')
for i in title_tags[0:10]:
    title=i.text
    Title.append(title)
```

```
rating_tags=driver.find_elements(By.XPATH, '//div[@class="a-row a-size-
small"]')
for i in rating_tags[0:10]:
    rating=i.text
    Rating.append(rating)
```

```
price_tags=driver.find_elements(By.XPATH, '//span[@class="a-price-whole"]')
for i in price_tags[0:10]:
    price=i.text
    Price.append(price)
```

In [103]:

```
print(len(Title),len(Rating),len(Price))
10 10 10
```



```
import pandas as pd
df=pd.DataFrame({"Title":Title,"Rating":Rating,"Price":Price})
df
```

In [104]:

Out[104]:

	Title	Rating	Price
0	Samsung Galaxy Book2 (NP750) Intel 12th Gen co...	4.1\n(102)	79,990
1	(Renewed) HP ELITEBOOK 840 G5 (Core i7 8th GEN...	New to Amazon	31,490
2	ASUS TUF Dash F15, Intel Core i7-12650H 12th G...	4.0\n(2)	92,990
3	ASUS TUF Gaming F15 (2022), 15.6"(39.62 cms) F...	4.4\n(71)	1,00,990
4	ASUS Vivobook S15 OLED 2022, 15.6" 39.62 cms F...	4.0\n(10)	84,990
5	LG Gram16 Intel EVO-[12th Gen Core i7/Win11/16...	4.1\n(55)	99,990
6	ASUS TUF Gaming F15 (2022), 15.6" (39.62 cms) ...	4.1\n(43)	94,990
7	(Renewed) HP 840g3 Elitebook 14 Inch Screen La...	3.0\n(2)	31,592
8	MI Notebook Horizon Edition 14 Intel Core i7-1...	4.1\n(1,187)	56,990
9	Lenovo ThinkPad E14 Intel Core i7 12th Gen 14"...	3.6\n(15)	98,990

In [ ]:

In [79]:

In [80]:

In [88]:

In [89]:

In [90]:

In [91]:

In [93]:

In [94]:

20 20 10

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In []:

In []:

In [38]:

In [39]:

In [40]:

In [44]:

In [45]:

In [47]:

In [49]:

In [50]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [1]:

In [2]:

In [3]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [57]:

In [63]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [14]:

In [15]:

In [16]:

In [18]:

In [19]:

In [20]:

In []:

In []:

In []:

In []:

In []:

In [13]:

In []:

In [4]:

In [5]:

In [7]:

In [8]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [31]:

In [32]:

In [34]:

In [35]:

In [36]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [25]:

In [35]:

In [38]:

In [40]:

In [111]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:



In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In [54]:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []:

In []: