```
pip install selenium

Requirement already satisfied: selenium in c:\users\dell\anaconda3\
lib\site-packages (4.17.2)
Requirement already satisfied: urllib3[socks]<3,>=1.26 in c:\users\
dell\anaconda3\lib\site-packages (from selenium) (1.26.16)
Requirement already satisfied: trio~=0.17 in c:\users\dell\anaconda3\
lib\site-packages (from selenium) (0.24.0)
Requirement already satisfied: trio-websocket~=0.9 in c:\users\dell\
anaconda3\lib\site-packages (from selenium) (0.11.1)
Requirement already satisfied: certifi>=2021.10.8 in c:\users\dell\
anaconda3\lib\site-packages (from selenium) (2023.7.22)
Requirement already satisfied: typing_extensions>=4.9.0 in c:\users\
dell\anaconda3\lib\site-packages (from selenium) (4.9.0)
Requirement already satisfied: attrs>=20.1.0 in c:\users\dell\
anaconda3\lib\site-packages (from trio~=0.17->selenium) (22.1.0)
Requirement already satisfied: sortedcontainers in c:\users\dell\
anaconda3\lib\site-packages (from trio~=0.17->selenium) (2.4.0)
Requirement already satisfied: idna in c:\users\dell\anaconda3\lib\
site-packages (from trio~=0.17->selenium) (3.4)
Requirement already satisfied: outcome in c:\users\dell\anaconda3\lib\
site-packages (from trio~=0.17->selenium) (1.3.0.post0)
Requirement already satisfied: sniffio>=1.3.0 in c:\users\dell\
anaconda3\lib\site-packages (from trio~=0.17->selenium) (1.3.0)
Requirement already satisfied: cffi>=1.14 in c:\users\dell\anaconda3\
lib\site-packages (from trio~=0.17->selenium) (1.15.1)
Requirement already satisfied: wsproto>=0.14 in c:\users\dell\
anaconda3\lib\site-packages (from trio-websocket~=0.9->selenium)
(1.2.0)
Requirement already satisfied: PySocks!=1.5.7,<2.0,>=1.5.6 in c:\
users\dell\anaconda3\lib\site-packages (from urllib3[socks]<3,>=1.26-
>selenium) (1.7.1)
Requirement already satisfied: pycparser in c:\users\dell\anaconda3\
lib\site-packages (from cffi>=1.14->trio~=0.17->selenium) (2.21)
Requirement already satisfied: h11<1,>=0.9.0 in c:\users\dell\
anaconda3\lib\site-packages (from wsproto>=0.14->trio-websocket~=0.9-
>selenium) (0.14.0)
Note: you may need to restart the kernel to use updated packages.

pip install webdriver_manager

Requirement already satisfied: webdriver_manager in c:\users\dell\
anaconda3\lib\site-packages (4.0.1)
Requirement already satisfied: requests in c:\users\dell\anaconda3\
lib\site-packages (from webdriver_manager) (2.31.0)
Requirement already satisfied: python-dotenv in c:\users\dell\
anaconda3\lib\site-packages (from webdriver_manager) (0.21.0)
Requirement already satisfied: packaging in c:\users\dell\anaconda3\
lib\site-packages (from webdriver_manager) (23.1)
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\
```

```
dell\anaconda3\lib\site-packages (from requests->webdriver_manager)
(2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\dell\
anaconda3\lib\site-packages (from requests->webdriver_manager) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\dell\
anaconda3\lib\site-packages (from requests->webdriver_manager)
(1.26.16)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\dell\
anaconda3\lib\site-packages (from requests->webdriver_manager)
(2023.7.22)
Note: you may need to restart the kernel to use updated packages.
```

```
import pandas as pd
from selenium import webdriver
from time import sleep
from selenium.webdriver.chrome.options import Options
from webdriver_manager.chrome import ChromeDriverManager
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

options=webdriver.ChromeOptions()
#q=input("Enter the query:")
driver=webdriver.Chrome(options=options)
page=driver.get('https://amazon.in')

a='Apple iphone'
search_bar = driver.find_element(By.ID,'twotabsearchtextbox')
search_bar.send_keys(a)
search_bar.send_keys(Keys.RETURN)
data=driver.find_elements(By.XPATH,'.//span[@class="a-size-medium a-
color-base a-text-normal"]')
names=[i.text for i in data]
pricdata=driver.find_elements(By.XPATH,'.//span[@class="a-price-
whole"]')
prices=[i.text for i in pricdata]
dataframe1=pd.DataFrame(columns=['name','price'])

names=[i.text for i in data]
pricdata=driver.find_elements(By.XPATH,'.//span[@class="a-price-
whole"]')
prices=[i.text for i in pricdata]
dataframe1=pd.DataFrame(columns=['name','price'])

for i in range(0,len(prices)):
    dataframe1.loc[i]=[names[i],prices[i]]
dataframe1
```

	name	price
0	Apple iPhone 13 (128GB) - Midnight	51,790
1	Apple iPhone 14 Pro Max (1 TB) - Gold	1,89,900
2	Apple iPhone 15 (256 GB) - Green	80,990

```
3                       Apple iPhone 13 (128GB) - Blue    51,790
4                   Apple iPhone 13 (128GB) - Midnight    51,790
5                      Apple iPhone 15 (128 GB) - Pink    71,490
6                    Apple iPhone 15 (256 GB) - Yellow    80,990
7       Apple iPhone 14 Plus (256 GB) - (Product) RED    77,999
8             Apple iPhone 13 (256GB) - (Product) RED    61,900
9    Apple iPhone 15 Pro Max (256 GB) - Blue Titanium  1,48,900
10                Apple iPhone 13 (128GB) - Starlight    51,790
11                  Apple iPhone 15 (128 GB) - Black    71,490
12                Apple iPhone 15 Plus (128 GB) - Blue    80,990
13                  Apple iPhone 15 (128 GB) - Green    71,490
14           Apple iPhone 14 (256 GB) - (Product) RED    65,998
15  Original Smartphone Compatible with Apple iPho...    10,999
16                     Apple iPhone 13 (256GB) - Blue    62,999
17                    Apple iPhone 15 (256 GB) - Pink    84,900
```

```
driver2=webdriver.Chrome(options=options)
page=driver2.get('https://flipKart.com')
search=driver2.find_element(By.XPATH,'.//input[@class="Pke_EE"]')
search.send_keys(a)
search.send_keys(Keys.RETURN)
name=driver2.find_elements(By.XPATH,'.//div[@class="_4rR01T"]')
names1=[i.text for i in name]

price=driver2.find_elements(By.XPATH,'.//div[@class="_30jeq3
_1_WHN1"]')
prices1=[i.text for i in price]
dataframe2=pd.DataFrame(columns=['name','price'])

for i in range(0,len(names1)):
    dataframe2[i]=[names1[i],prices1[i]]
dataframe2
```

```
  name price                                      0
1  \
0  NaN   NaN  Apple iPhone 15 (Blue, 128 GB)  Apple iPhone 14 (Blue,
128 GB)
1  NaN   NaN                          ₹72,999
₹58,999

                                          2
3  \
0  Apple iPhone 13 (Starlight, 128 GB)  Apple iPhone 14 (Starlight,
128 GB)
1                                ₹52,999
₹58,999

                                         4                                    5
\
0  Apple iPhone 14 (Midnight, 128 GB)  Apple iPhone 13 (Green, 128 GB)
```

```
1                                ₹58,999                                ₹52,999


                                      6
7  ...  \
0  Apple iPhone 14 (Purple, 128 GB)  Apple iPhone 13 (Pink, 128
GB)  ...
1                               ₹58,999
₹52,999  ...

                                     14                                     15  \
0  Apple iPhone 14 (Blue, 256 GB)  Apple iPhone 12 (White, 64 GB)
1                              ₹68,999                                ₹44,999

                                    16
17  \
0  Apple iPhone 12 (Blue, 64 GB)  Apple iPhone 14 Plus (Starlight, 128
GB)
1                             ₹44,999
₹66,999

                                           18  \
0  Apple iPhone 14 Plus (Purple, 128 GB)
1                                  ₹66,999

                                            19  \
0  Apple iPhone 14 ((PRODUCT)RED, 128 GB)
1                                   ₹58,999

                                         20
21  \
0  Apple iPhone 15 Plus (Blue, 128 GB)  Apple iPhone 15 (Yellow, 128
GB)
1                                 ₹82,999
₹72,999

                                          22
23
0  Apple iPhone 15 Plus (Black, 128 GB)  Apple iPhone 14 (Starlight,
256 GB)
1                                  ₹82,999
₹68,999

[2 rows x 26 columns]
```

```
product1,product2=[],[]
for i in range(11):
    count=0
    for char in a.split(' '):
```

```
        if char.lower() in names1[i].lower():
            count=count+1
    if count>=len(a.split(' ')):
        product2.append((names1[i],prices1[i]))
for i in range(11):
    count=0
    for char in a.split(' '):
        if char.lower() in names[i].lower():
            count=count+1
    if count>=len(a.split(' ')):
        product1.append((names[i],prices[i]))
for pro in product2:
    print(pro)
```

```
('Apple iPhone 15 (Blue, 128 GB)', '₹72,999')
('Apple iPhone 14 (Blue, 128 GB)', '₹58,999')
('Apple iPhone 13 (Starlight, 128 GB)', '₹52,999')
('Apple iPhone 14 (Starlight, 128 GB)', '₹58,999')
('Apple iPhone 14 (Midnight, 128 GB)', '₹58,999')
('Apple iPhone 13 (Green, 128 GB)', '₹52,999')
('Apple iPhone 14 (Purple, 128 GB)', '₹58,999')
('Apple iPhone 13 (Pink, 128 GB)', '₹52,999')
('Apple iPhone 13 (Midnight, 128 GB)', '₹52,999')
('Apple iPhone 15 (Green, 128 GB)', '₹72,999')
('Apple iPhone 13 (Blue, 128 GB)', '₹52,999')
```

```
if int(product1[0][1][1:].replace(',',''))>int(product2[0][1]
[1:].replace(',','')):
    print("flipKart has best prices")
else:
    print('amazon has best prices')
```

```
amazon has best prices
```