In [ ]:
```
!pip install --upgrade selenium
```

In [ ]:
```
!pip install webdriver_manager selenium
```

1. All the questions must be done in a single Jupyter notebook.

2. There should be proper comments incode.

Q1: Write a python program to scrape data for “Data Analyst” Job position in “Bangalore” location. You

have to scrape the job-title, job-location, company_name, experience_required. You have to scrape first 10 jobs data. This task will be done in following steps:

1. First get the webpage https://www.shine.com/
2. Enter “Data Analyst” in “Skill, Designations, Companies” field and enter “Bangalore” in “enter the location” field.
3. Then click the search button.
4. Then scrape the data for the first 10 jobs results you get.
5. Finally create a dataframe of the scraped data. Note: All of the above steps have to be done in code. No step is to be done manually.

In [3]:
```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
import time
from selenium.webdriver.common.by import By

# Creates Chrome WebDriver instance.
driver = webdriver.Chrome()

# Let's maximize the automated chrome window
driver.maximize_window()

# Navigate to shine website
driver.get("https://www.shine.com/")

# Wait for the page to load
time.sleep(10)

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

search_field_designation = driver.find_element(By.XPATH, '//*[@id="webSearchBar"]/span')
search_field_designation.click()

time.sleep(5)

Job_title = driver.find_element(By.XPATH, '//*[@id="id_q"]')
Job_title.send_keys('Data Analyst')

location = driver.find_element(By.XPATH, '//*[@id="id_loc"]')
location.send_keys('Bangalore')
```

```
search_job = driver.find_element(By.XPATH, '//*[@id="__next"]/div[4]/div/div[2]/div[2]/div/form/div/div[2]/div/button')
search_job.click()
time.sleep(15)

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

# Initialize empty lists to store data
job_titles = []
company_names = []
locations = []
experiences = []

# Scrape data from the first 10 listings
while len(job_titles) < 10:

    # Find all the products on the page //*[@id="__next"]/div[1]/div[5]/div/div[1]/div[1]/div
    product_tags = driver.find_elements(By.CLASS_NAME, "jobCard_jobCard__jjUmu")

    # Loop through each product and extract the required attributes
    for product in product_tags:


        # Get job title
        try:
            job_title = product.find_element(By.TAG_NAME, "h2").text
            job_titles.append(job_title)
        except:
            job_titles.append("")

        try:
            company_name = product.find_element(By.CLASS_NAME, "jobCard_jobCard_cName__mYnow").text
            company_names.append(company_name)
        except:
            job_titles.append("")

        # Get location data
        try:
            loca = product.find_element(By.CLASS_NAME, "jobCard_locationIcon__zrWt2")
            locations.append(loca.text.split('\n')[0])
        except:
            locations.append("")

        # Get experience data
        try:
            experi = product.find_element(By.CLASS_NAME, "jobCard_jobIcon__3FB1t").text
            experiences.append(experi)
        except:
            experiences.append("")

        # If we haven't collected enough data, go to the next page
        if len(job_titles) >= 10:
            break

# Create a DataFrame
data = pd.DataFrame({'Job Title': job_titles, 'Company Name': company_names,'Locations': locations, 'Experiences': experiences})

# Display the DataFrame
display(data)
```

```
# Close the ChromeDriver
driver.quit()
```

	Job Title	Company Name	Locations	Experiences
0	Data Analyst	quess corp (magna infotech)	Bangalore	2 to 5 Yrs
1	Require for the Data Analytics _Senior Analyst	acura solutions.	Bangalore	4 to 6 Yrs
2	Data _ Analytics Senior __Analyst	acura solutions.	Bangalore	4 to 6 Yrs
3	Clinical Data Analyst	quiscon biotech	Bangalore	0 to 2 Yrs
4	Sr. Data Analyst (SQL) - Pharma	ara resources private limited	Bangalore	5 to 8 Yrs
5	Data Analyst, FCGT	amazon	Bangalore	1 to 3 Yrs
6	Clinical Data Analyst	radiometer india	Bangalore	4 to 6 Yrs
7	Lead Clinical Data Analyst	radiometer india	Bangalore	8 to 10 Yrs
8	Data analyst, Amazon Pay - RSO- GTM	amazon	Bangalore	2 to 5 Yrs
9	Data Analyst for ADAS effectiveness	mercede	Bangalore	2 to 5 Yrs

Q2: Write a python program to scrape data for “Data Scientist” Job position in “Bangalore” location. You

have to scrape the job-title, job-location, company_name. You have to scrape first 10 jobs data. This task will be done in following steps:

1. First get the webpage https://www.shine.com/
2. Enter “Data Scientist” in “Skill, Designations, Companies” field and enter “Bangalore” in “enter the location” field.
3. Then click the search button.
4. Then scrape the data for the first 10 jobs results youget.
5. Finally create a dataframe of the scraped data. Note: All of the above steps have to be done in code. No step is to be done manually.

In [7]:
```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Key

# Creates Chrome WebDriver instance.
driver = webdriver.Chrome()

# Let's maximize the automated chrome window
driver.maximize_window()

# Navigate to shine website
driver.get("https://www.shine.com/")

# Wait for the page to load
time.sleep(10)

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass
```

```python
search_field_designation = driver.find_element(By.XPATH, '//*[@id="webSearchBar"]/span')
search_field_designation.click()

time.sleep(5)

Job_title = driver.find_element(By.XPATH, '//*[@id="id_q"]')
Job_title.send_keys('Data Scientist')

location = driver.find_element(By.XPATH, '//*[@id="id_loc"]')
location.send_keys('Bangalore')

search_job = driver.find_element(By.XPATH, '//*[@id="__next"]/div[4]/div/div[2]/div[2]/div/form/div/div[2]/div/button')
search_job.click()
time.sleep(15)

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

# Initialize empty lists to store data
job_titles = []
company_names = []
locations = []

# Scrape data from the first 10 listings
while len(job_titles) < 10:

    # Find all the products on the page
    product_tags = driver.find_elements(By.CLASS_NAME, "jobCard_jobCard__jjUmu")

    # Loop through each product and extract the required attributes
    for product in product_tags:


        # Get job title
        try:
            job_title = product.find_element(By.TAG_NAME, "h2").text
            job_titles.append(job_title)
        except:
            job_titles.append("")

            # Get company_name data
        try:
            company_name = product.find_element(By.CLASS_NAME, "jobCard_jobCard_cName__mYnow").text
            company_names.append(company_name)
        except:
            job_titles.append("")

        # Get location data
        try:
            loca = product.find_element(By.CLASS_NAME, "jobCard_locationIcon__zrWt2")
            locations.append(loca.text.split('\n')[0])
        except:
            locations.append("")

        # If we haven't collected enough data, go to the next page
        if len(job_titles) >= 10:
            break

# Create a DataFrame
```

```
data = pd.DataFrame({'Job Title': job_titles, 'Company Name': company_names,'Locations': locations})

# Display the DataFrame
display(data)

# Close the ChromeDriver
driver.quit()
```

	Job Title	Company Name	Locations
0	Data Scientist	skyleaf consultants	Bangalore
1	Data Scientist	seven geomax consulting private lim...	Bangalore
2	Hiring now Data Scientist	skyleaf consultants	Bangalore
3	Data Scientist	quiscon biotech	Bangalore
4	Req. now Data Scientist	infosys limited	Bangalore
5	Data Scientist	acme services private limited	Bangalore
6	Data Scientist	acuity knowledge partners	Bangalore
7	Data Scientist Urgent Recruitment	niharika enterprises	Bangalore
8	DTICI Data Scientists - Senior Consultant - T8	mercede	Bangalore
9	Data Scientist - Supply Chain Analytics	ntt global delivery services limite...	Bangalore

Q3:You have to use the location and salary filter.

You have to scrape data for “Data Scientist” designation for first 10 job results. You have to scrape the job-title, job-location, company name, experience required. The location filter to be used is “Bangalore”. The salary filter to be used is “3-6” lakhs The task will be done as shown in the below steps:

1. first get thewebpage https:https://www.shine.com/
2. Enter “Data Scientist” in “Skill, Designations, and Companies” field.
3. Then click the searchbutton.
4. Then apply the location filter and salary filter by checking the respective boxes
5. Then scrape the data for the first 10 jobs results youget.
6. Finally create a dataframe of the scraped data. Note: All of the above steps have to be done in code. No step is to be done manually.

In [11]:
```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Key

# Creates Chrome WebDriver instance.
driver = webdriver.Chrome()

# Let's maximize the automated chrome window
driver.maximize_window()

# Navigate to shine website
driver.get("https://www.shine.com/")

# Wait for the page to load
time.sleep(10)
```

```
# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

search_field_designation = driver.find_element(By.XPATH, '//*[@id="webSearchBar"]/span')
search_field_designation.click()

time.sleep(5)

Job_title = driver.find_element(By.XPATH, '//*[@id="id_q"]')
Job_title.send_keys('Data Scientist')

location = driver.find_element(By.XPATH, '//*[@id="id_loc"]')
location.send_keys('Bangalore')

search_job = driver.find_element(By.XPATH, '//*[@id="__next"]/div[4]/div/div[2]/div[2]/div/form/div/div[2]/div/button')
search_job.click()
time.sleep(15)

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

salary = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[4]/div/div[1]/div/div[2]/div/ul/li[3]/button')
salary.click()

time.sleep(3)
salary1  = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[4]/div/div[1]/div/div[2]/div[2]/div/div/div/div[3]/div/div/div/ul/li[3]/span/label')
salary1.click()

salary2 = driver.find_element(By.XPATH, '//*[@id="__next"]/div[1]/div[4]/div/div[1]/div/div[2]/div[2]/div/div/div/div[4]/button[2]')
salary2.click()

# Initialize empty lists to store data
job_titles = []
company_names = []
locations = []
experiences = []

# Scrape data from the first 10 listings
while len(job_titles) < 10:

    # Find all the products on the page
    product_tags = driver.find_elements(By.CLASS_NAME, "jobCard_jobCard__jjUmu")

    # Loop through each product and extract the required attributes
    for product in product_tags:


        # Get job title
        try:
            job_title = product.find_element(By.TAG_NAME, "h2").text
            job_titles.append(job_title)
        except:
            job_titles.append("")

            # Get company_name data
        try:
```

```
            company_name = product.find_element(By.CLASS_NAME, "jobCard_jobCard_cName__mYnow").text
            company_names.append(company_name)
        except:
            job_titles.append("")

        # Get location data
        try:
            loca = product.find_element(By.CLASS_NAME, "jobCard_locationIcon__zrWt2")
            locations.append(loca.text.split('\n')[0])
        except:
            locations.append("")

        # Get experience data
        try:
            experi = product.find_element(By.CLASS_NAME, "jobCard_jobIcon__3FB1t").text
            experiences.append(experi)
        except:
            experiences.append("")

        # If we haven't collected enough data, go to the next page
        if len(job_titles) >= 10:
            break

# Create a DataFrame
data = pd.DataFrame({'Job Title': job_titles, 'Company Name': company_names,'Locations': locations, 'Experiences': experiences})

# Display the DataFrame
display(data)

# Close the ChromeDriver
driver.quit()
```

	Job Title	Company Name	Locations	Experiences
0	Data Scientist	skyleaf consultants	Bangalore	5 to 10 Yrs
1	Data Scientist	quiscon biotech	Bangalore	0 to 1 Yr
2	Req. now Data Scientist	infosys limited	Bangalore	0 to 5 Yrs
3	Data Scientist	acme services private limited	Bangalore	3 to 5 Yrs
4	Data Scientist Urgent Recruitment	niharika enterprises	Bangalore	0 to 4 Yrs
5	DTICI Data Scientists - Senior Consultant - T8	mercede	Bangalore	3 to 7 Yrs
6	Data Scientist	skyleaf consultants	Bangalore	5 to 10 Yrs
7	Sr. Data Engineer	ibm india pvt. limited	Bangalore	2 to 8 Yrs
8	Impact Analytics - Senior Data Scientist	impact analytics	Bangalore	2 to 4 Yrs
9	DTICI Senior Consultant - Data Scientist - T8	mercede	Bangalore	3 to 7 Yrs

Q4: Scrape data of first 100 sunglasses listings on flipkart.com. You have to scrape four attributes:

1. Brand
2. Product Description
3. Price

To scrape the data you have to go through following steps:

1. Go to Flipkart webpage by url : https://www.flipkart.com/

2. Enter “sunglasses” in the search field where “search for products, brands and more” is written and click the search icon
3. After that you will reach to the page having a lot of sunglasses. From this page you can scrap the required data as usual.
4. After scraping data from the first page, go to the “Next” Button at the bottom other page , then click on it.
5. Now scrape data from this page as usual
6. Repeat this until you get data for 100 sunglasses.

Note: That all of the above steps have to be done by coding only and not manually.

In [12]:
```
import selenium     #library that is used to work with selenium
import pandas as pd    #to create DataFrame
from selenium import webdriver    #importing webdriver module from selenium to open automated chrome window
import warnings
warnings.filterwarnings('ignore')  #to ignore any sort of warning
from selenium.webdriver.common.by import By   #importing inbuilt class By
import time    #use to stop search engine for few seconds
from webdriver_manager.chrome import ChromeDriverManager #Automatically manages Chrome WebDriver installation.
from selenium.webdriver.common.keys import Keys #Simulates keyboard events in Selenium.

#Creates Chrome WebDriver instance.
driver = webdriver.Chrome()

#Let's maximize the automated chrome window
driver.maximize_window()

# Navigate to Flipkart website
driver.get("https://www.flipkart.com/")

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '/html/body/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

# Search for sunglasses
search_input = driver.find_element(By.XPATH,'//*[@id="container"]/div/div[1]/div[1]/div[2]/div[2]/form/div/div/input')
search_input.send_keys('Sun glasses')
search_input.send_keys(Keys.ENTER)

# Wait for the page to load
time.sleep(3)

# Initialize empty lists to store data
brand_names = []
product_descriptions = []
prices = []
discounts = []


# Scrape data from the first 100 listings
while len(brand_names) < 100:

    # Find all the products on the page
    product_tags = driver.find_elements(By.CLASS_NAME, "_2B099V")
    # Loop through each product and extract the required attributes

    for product in product_tags:

        # Get brand name
            try:
                brand = product.find_element(By.CLASS_NAME, "_2WkVRV").text
```

```
            brand_names.append(brand)
        except:
            brand_names.append("")

        # Get product description
        try:
            description = product.find_element(By.CLASS_NAME, "IRpwTa").text
            product_descriptions.append(description)
        except:
            product_descriptions.append("")
        # Get price
        try:
            price = product.find_element(By.CLASS_NAME, "_30jeq3").text
            prices.append(price)
        except:
            prices.append("")
        #Get discount
        try:
            discount = product.find_element(By.CLASS_NAME,  "_3Ay6Sb").text
            discounts.append(discount)
        except:
            discounts.append("")

        #If we haven't collected enough data, go to the next page
        if len(brand_names) >= 99:
             break
try:
    next_button = driver.find_elements(By.CLASS_NAME,  "_1LKTO3")
    next_button.pop().click()
            # Wait for the next page to load
    time.sleep(3)
except selenium.common.exceptions.NoSuchElementException:
        pass # If there is no "Next" button, exit the loop

data = pd.DataFrame({'Brand': brand_names,'Product Description': product_descriptions,'Price': prices, 'Discount': discounts})

# Display the DataFrame
display(data)

# Close the ChromeDriver
driver.quit()
```

	Brand	Product Description	Price	Discount
0	VOLRIX	UV Protection Rectangular Sunglasses (Free Size)	₹263	82% off
1	ROYAL SON	UV Protection Rectangular, Retro Square Sungla...	₹497	66% off
2	Fastrack	UV Protection Rectangular Sunglasses (Free Size)	₹675	24% off
3	SRPM	UV Protection Wayfarer Sunglasses (50)	₹194	85% off
4	Elligator	UV Protection Cat-eye, Retro Square, Oval, Rou...	₹168	71% off
...	...	...	...	...
95	Sukart	UV Protection, Polarized, Gradient, Riding Gla...	₹292	70% off
96	VINCENT CHASE	by Lenskart Polarized, UV Protection Wayfarer ...	₹693	65% off
97	ROYAL SON	Polarized, UV Protection Retro Square Sunglass...	₹664	55% off
98	ROZZETTA CRAFT	Polarized, Riding Glasses Sports, Wrap-around ...	₹480	75% off
99	VINCENT CHASE	Polarized, UV Protection Aviator Sunglasses (56)	₹854	57% off

100 rows × 4 columns

Q5: Scrape 100 reviews data from flipkart.com for iphone11 phone. You have to go the link:

https://www.flipkart.com/apple-iphone-11-black-64-gb/product-reviews/itm4e5041ba101fd?pid=MOBFWQ6BXGJCEYNY&lid=LSTMOBFWQ6BXGJCEYNYZXSHRJ&marketplace=FLIPKART

As shown in the above page you have to scrape the tick marked attributes. These are:

1. Rating
2. Review summary
3. Full review
4. You have to scrape this data for first 100reviews. Note: All the steps required during scraping should be done through code only and not manually.

In [13]:
```
import selenium     #library that is used to work with selenium
import pandas as pd    #to create DataFrame
from selenium import webdriver    #importing webdriver module from selenium to open automated chrome window
import warnings
warnings.filterwarnings('ignore')  #to ignore any sort of warning
from selenium.webdriver.common.by import By   #importing inbuilt class By
import time    #use to stop search engine for few seconds

from webdriver_manager.chrome import ChromeDriverManager #Automatically manages Chrome WebDriver installation.

from selenium.webdriver.common.keys import Keys #Simulates keyboard events in Selenium.

driver = webdriver.Chrome()

driver.maximize_window()

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

time.sleep(3)

# Initialize empty lists to store review data
ratings = []
review_summaries = []
full_reviews = []
```

```
# Scrape data from the first 100 reviews
while len(ratings) < 100:

     review_tags = driver.find_elements(By.CLASS_NAME, "_2wzgFH")

     for review in review_tags:
        rating = review.find_element(By.CLASS_NAME, "_3LWZlK").text #if rating else '3'
        ratings.append(rating)

        review_summary =review.find_element(By.CLASS_NAME, "_2-N8zT").text #if review_summary  else '2'
        review_summaries.append(review_summary)

        full_review = review.find_element(By.CLASS_NAME, "t-ZTKy").text  #if full_review else "1"
        full_reviews.append(full_review)
        if len(ratings) >= 100:
            break

    #If we haven't collected enough data, go to the next page
     if len(ratings) > 99:
        try:
            next_button = driver.find_element(By.XPATH, '//*[@id="container"]/div/div[3]/div/div/div[2]/div[13]/div/div/nav/a[11]')
            next_button.click()
            # Wait for the next page to load
            time.sleep(3)
        except selenium.common.exceptions.NoSuchElementException:
            break  # If there is no "Next" button, exit the loop


data = pd.DataFrame({'Rating': ratings,'Review Summary': review_summaries, 'Full Review': full_reviews})

# Display the DataFrame
display(data)

# Close the ChromeDriver
driver.quit()
```

	Rating	Review Summary	Full Review
0	5	Classy product	Camera is awesome\nBest battery backup\nA perf...
1	5	Mind-blowing purchase	Photos super
2	5	Wonderful	This is amazing at all
3	5	Terrific	Very very good
4	5	Fabulous!	Super 🔥 and good performance 👌 ❤️
...	...	...	...
95	5	Just wow!	Perfect Product!!
96	5	Must buy!	It’s really awesome
97	5	Perfect product!	V Good all
98	5	Best in the market!	Good Camera
99	5	Worth every penny	Feeling awesome after getting the delivery of ...

100 rows × 3 columns

Q6: Scrape data for first 100 sneakers you find when you visit flipkart.com and search for “sneakers” in the

search field. You have to scrape 3 attributes of each sneaker:

1. Brand
2. Product Description
3. Price

In [15]:
```
import selenium     #library that is used to work with selenium
import pandas as pd    #to create DataFrame
from selenium import webdriver    #importing webdriver module from selenium to open automated chrome window
import warnings
warnings.filterwarnings('ignore')  #to ignore any sort of warning
from selenium.webdriver.common.by import By   #importing inbuilt class By
import time    #use to stop search engine for few seconds
from webdriver_manager.chrome import ChromeDriverManager #Automatically manages Chrome WebDriver installation.
from selenium.webdriver.common.keys import Keys #Simulates keyboard events in Selenium.

 #Creates Chrome WebDriver instance.
driver = webdriver.Chrome()

#Let's maximize the automated chrome window
driver.maximize_window()

# Navigate to Flipkart website
driver.get("https://www.flipkart.com/")

# Close the pop-up dialog (if it appears)
try:
    x = driver.find_element(By.XPATH, '/html/body/div[2]/div/div/button')
    x.click()
except selenium.common.exceptions.NoSuchElementException:
    pass

# Search for sneakers
search_input = driver.find_element(By.XPATH,'//*[@id="container"]/div/div[1]/div[1]/div[2]/div[2]/form/div/div/input')
search_input.send_keys('sneakers')
search_input.send_keys(Keys.ENTER)

# Wait for the page to load
time.sleep(3)

# Initialize empty lists to store data
brand_names = []
product_descriptions = []
prices = []
discounts = []


# Scrape data from the first 100 listings
while len(brand_names) < 100:

    # Find all the products on the page
    product_tags = driver.find_elements(By.CLASS_NAME, "_2B099V")
    # Loop through each product and extract the required attributes

    for product in product_tags:

        # Get brand name
            try:
                brand = product.find_element(By.CLASS_NAME, "_2WkVRV").text
                brand_names.append(brand)
            except:
                brand_names.append("")
```

```
            # Get product description
            try:
                description = product.find_element(By.CLASS_NAME, "IRpwTa").text
                product_descriptions.append(description)
            except:
                product_descriptions.append("")
            # Get price
            try:
                price = product.find_element(By.CLASS_NAME, "_30jeq3").text
                prices.append(price)
            except:
                prices.append("")
            #Get discount
            try:
                discount = product.find_element(By.CLASS_NAME,  "_3Ay6Sb").text
                discounts.append(discount)
            except:
                discounts.append("")

            #If we haven't collected enough data, go to the next page
            if len(brand_names) >= 99:
                 break
    try:
        next_button = driver.find_elements(By.CLASS_NAME,  "_1LKTO3")
        next_button.pop().click()
                # Wait for the next page to load
        time.sleep(3)
    except selenium.common.exceptions.NoSuchElementException:
        pass # If there is no "Next" button, exit the loop

data = pd.DataFrame({'Brand': brand_names,'Product Description': product_descriptions,'Price': prices, 'Discount': discounts})

# Display the DataFrame
display(data)

# Close the ChromeDriver
driver.quit()
```

	Brand	Product Description	Price	Discount
0	BIRDE	Combo Pack of 2 Casual Shoes Sneakers For Men	₹479	76% off
1	Airson	Junior Zero1 Sports shoes for Men | Gym Traini...	₹989	55% off
2	Magnolia	Modern Trendy Sneakers boot Sneakers Sneakers ...	₹499	61% off
3	BRUTON	Modern Trendy Shoes Sneakers For Men	₹299	76% off
4	aadi	Lightweight,Comfort,Summer,Trendy,Walking,Outd...	₹389	80% off
...	...	...	...	...
95	HRX by Hrithik Roshan	Club Culture Sneakers For Men	₹1,047	73% off
96	HIGHLANDER	Sneakers For Men	₹696	65% off
97	Kraasa	Sneakers For Women	₹499	50% off
98	BIRDE	Premium Casual Shoes For Men Sneakers For Men	₹479	76% off
99	K- FOOTLANCE	Casual Sneakers White Outdoor Shoes For Boys A...	₹549	45% off

100 rows × 4 columns

In [ ]: