Using `Selenium` for web scraping

Example 5. Product price scraping from https://www.daraz.com.np/smartphones

---

In [1]:
```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.firefox.service import Service as FirefoxService
from webdriver_manager.firefox import GeckoDriverManager
from lxml import html
import pandas as pd
import time
from io import StringIO
```

In [2]:
```
driver = webdriver.Firefox(service=FirefoxService(GeckoDriverManager().install()))
driver.get("https://www.daraz.com.np/smartphones")

product = []
rating = []
rating_no = []
sales = []
price = []

def web_scrap():
    tree = html.fromstring(driver.page_source)
    elems = tree.xpath("//div[starts-with(@class,'description')]")
    for e in elems:
        val = e.xpath("div[1]")
        product.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[2]//span[2]")
        rating.append(val[0].text_content().strip() if len(val) > 0 else '')
```

```
        val = e.xpath("div[2]//span[3]")
        rating_no.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[2]/div[3]")
        sales.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[@id='id-price']//div[starts-with(@class,'current-price')]")
        price.append(val[0].text_content().strip() if len(val) > 0 else '')
```

In [3]:
```
#navigating pages from 1 to 3 and scraping data
for i in range(3):
    e = driver.find_element(By.XPATH, f"//li[@title = '{i+1}' ]")
    e.click()
    web_scrap()
```

In [4]:
```
df = pd.DataFrame({'product': product, 'rating':rating, 'rating_no':rating_no, 'sales':sales, 'price':price})
display(df.head())

df.to_csv('example5.csv', index=False)
```

	product	rating	rating_no	sales	price
0	Tecno Spark 20 Pro+ (16*/256 GB) | 6.78" FHD +...	4.5/5	(40)	127 Sold	Rs.26,990
1	Redmi Note 11 | 90 Hz FHD+ AMOLED Display | 50...	4.5/5	(115)	377 Sold	Rs.23,999
2	Redmi 13C (6/128GB) | 6.74" Dot Drop display |...	4.4/5	(36)	156 Sold	Rs.15,999
3	realme C53 (6+128 GB) | 6.74 inch HD+ IPS LCD ...	4.1/5	(18)	85 Sold	Rs.16,499
4	Redmi Note 13 Pro (8/256GB) | 6.67" AMOLED Dis...	4.3/5	(19)	80 Sold	Rs.32,999

Practice 3. From https://www.sharesansar.com/today-share-price, scrape stock data of Commercial Bank from date 2024-06-06 to 2024-06-11

In [5]:
```
#loading the website
driver.get("https://www.sharesansar.com/today-share-price")
```

In [6]:
```
#clicking on the dropdown box of sector
driver.find_element(By.XPATH, "//span[@id='select2-sector-container']").click()

#Finding input field to type
e = driver.find_element(By.XPATH, "//input[@role='textbox' and @type='search']")
e.clear()
e.send_keys('Commercial Bank')
e.send_keys(Keys.ENTER)

#list of date to scrape
lst_date = ['2024-06-06','2024-06-07','2024-06-08','2024-06-09', '2024-06-10', '2024-06-11']

#removing any previous dataframe named df_stock
if 'df_stock' in locals():
    del df_stock

for l in lst_date:
    e = driver.find_element(By.XPATH, "//input[@name='date']")
    e.clear()
    e.send_keys(l)
    e.send_keys(Keys.ENTER)

    driver.find_element(By.XPATH, '//button[@id="btn_todayshareprice_submit"]').click()
    time.sleep(3)

    webpage = driver.page_source #obtaining html code from the page

    if 'No Record Found.' not in webpage: #checking whether No record found is displayed in the page or not
        dd = pd.read_html(StringIO(webpage))[0]
        dd['date_en'] = l #adding a date column

        if 'df_stock' in locals():
            df_stock = pd.concat([df_stock, dd])
        else:
            df_stock = dd
```

In [7]:
```
display(df_stock.head())
df_stock.to_csv("practice3.csv", index=False)
```

	S.No	Symbol	Conf.	Open	High	Low	Close	VWAP	Vol	Prev. Close	Turnover	Trans.	Diff	Range	Diff %	Range %	VWAP %	52 Weeks High
0	1	ADBL	44.64	268.5	268.5	260.0	261.9	260.75	23243.0	264.0	6060520.5	168	-2.1	8.5	-0.80	3.27	0.44	292.9
1	2	CZBIL	39.02	167.1	168.0	165.2	167.0	166.06	29887.0	168.0	4963024.6	83	-1.0	2.8	-0.60	1.69	0.56	207.8
2	3	EBL	45.35	528.1	528.1	521.2	528.0	524.52	27100.0	528.9	14214538.1	144	-0.9	6.9	-0.17	1.32	0.66	633.0
3	4	GBIME	39.92	183.0	184.0	180.1	180.5	180.89	64245.0	183.9	11621120.2	340	-3.4	3.9	-1.85	2.17	-0.21	241.9
4	5	HBL	41.49	190.0	190.0	185.0	187.5	186.92	27461.0	189.5	5133072.2	134	-2.0	5.0	-1.06	2.70	0.31	240.0

Practice 4. From https://www.daraz.com.np search for top selling `rice` products. Then, scrape rice prices from 1 to 5 pages

In [8]:
```
#loading the website
driver.get("https://www.daraz.com.np")

product = []
rating = []
rating_no = []
sales = []
current_price = []
original_price = []
```

```
def web_scrap():
    tree = html.fromstring(driver.page_source)
    elems = tree.xpath("//div[starts-with(@class,'description')]")
    for e in elems:
        val = e.xpath("div[1]")
        product.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[2]//span[2]")
        rating.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[2]//span[3]")
        rating_no.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[2]/div[3]")
        sales.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[@id='id-price']//div[starts-with(@class,'current-price')]")
        current_price.append(val[0].text_content().strip() if len(val) > 0 else '')

        val = e.xpath("div[@id='id-price']//div[starts-with(@class,'original-price')]")
        original_price.append(val[0].text_content().strip() if len(val) > 0 else '')

#searching for rice products
e = driver.find_element(By.XPATH, "//input[@id='q']")
e.clear()
e.send_keys('rice')
e.send_keys(Keys.ENTER)
time.sleep(5)

#sort by Top Sales
driver.find_element(By.XPATH, "//div[@role='combobox']").click()
driver.find_element(By.XPATH, "//li[@title='Top Sales']").click()
```

In [9]:

```
#navigating pages from 1 to 5 and scraping data
for i in range(5):
    e = driver.find_element(By.XPATH, f"//li[@title = '{i+1}' ]")
    e.click()
    web_scrap()
    time.sleep(2)
```

In [10]:
```
df = pd.DataFrame({'product': product, 'rating':rating, 'rating_no':rating_no, 'sales':sales, 'current_price':current_price, '
display(df.head())

df.to_csv('practice4.csv', index=False)
```

	product	rating	rating_no	sales	current_price	original_price
0	Lal Qilla Brown Basmati Rice 1 kg	4.4/5	(7)	31 Sold	Rs.410	Rs. 500
1	DhikiJato Local Anadi Chamal 1 KG	4.5/5	(20)	145 Sold	Rs.350	
2	Dhiki Jato Jumla Marsi Chamal 1kg	4.4/5	(10)	51 Sold	Rs.270	
3	Newari Shahi Pulao Basmati Rice 5 Kg	4.9/5	(14)	94 Sold	Rs.995	
4	Taichin Chamal 1Kg	4.7/5	(12)	161 Sold	Rs.195	

In [11]:
```
driver.quit()
```