

In [1]: `!pip install selenium`

```
Requirement already satisfied: selenium in c:\users\ebele okonkwo\anaconda3\lib\site-packages (4.16.0)
Requirement already satisfied: urllib3[socks]<3,>=1.26 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from selenium) (1.26.16)
Requirement already satisfied: trio~=0.17 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from selenium) (0.24.0)
Requirement already satisfied: trio-websocket~=0.9 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from selenium) (0.11.1)
Requirement already satisfied: certifi>=2021.10.8 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from selenium) (2023.11.17)
Requirement already satisfied: attrs>=20.1.0 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (22.1.0)
Requirement already satisfied: sortedcontainers in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (2.4.0)
Requirement already satisfied: idna in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (3.4)
Requirement already satisfied: outcome in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (1.3.0.post0)
Requirement already satisfied: sniffio>=1.3.0 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (1.3.0)
Requirement already satisfied: cffi>=1.14 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from trio~=0.17->selenium) (1.15.1)
Requirement already satisfied: wsproto>=0.14 in c:\users\ebele okonkwo\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\ebele okonkwo\anaconda3\lib\site-packages (from urllib3[socks]<3,>=1.26->selenium) (1.7.1)
Requirement already satisfied: pycparser in c:\users\ebele okonkwo\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\ebele okonkwo\anaconda3\lib\site-packages (from wsproto>=0.14->trio-websocket~=0.9->selenium) (0.12.0)
```

In [23]: `import selenium
import pandas as pd
import time
from selenium import webdriver
from selenium.common.exceptions import StaleElementReferenceException, NoSuchElementException
import requests
from selenium.webdriver.common.by import By`

In [24]: `driver = webdriver.Chrome()`

In [25]: `driver.get('https://www.amazon.in/')`



```
In [37]: for k in name:
          name.append(k.text)
          name[:100]
```

```
In [38]: for l in price:
          price.append(l.text)
          price[:100]
```

```
In [39]: for m in return_exchange:
          return_exchange.append(m.text)
          return_exchange[:100]
```

```
In [40]: for n in expected_delivery:
          expected_delivery.append(n.text)
          expected_delivery[:100]
```

```
In [18]: for o in availability:
          availability.append(o.text)
          availability[:100]
```

```
In [41]: for o in product_url:
          product_url.append(o.text)
          product_url[:100]
```

```
In [42]: print(len(name[:100])),print(len(price[:100])),print(len(return_exchange[:100])),print(len(expected_delivery[:100])),print(len(availability[:100])),print(len(product_url[:100]))
```

```
0
0
0
0
0
0
0
```

```
Out[42]: (None, None, None, None, None, None)
```

```
In [43]: Guitars=pd.DataFrame({'Brand Name': brand,'Name of the Product': name,'Price': price,'Return/Exchange': return_exchange,'Expected Delivery': expected_delivery,'Availability': availability,'Product URL': product_url})
```

```
In [44]: Guitars
```

```
Out[44]:
```

Brand Name	Name of the Product	Price	Return/Exchange	Expected Delivery	Availability	Product URL
------------	---------------------	-------	-----------------	-------------------	--------------	-------------



```
In [86]: cars = driver.find_element(By.XPATH, '/html/body/div[1]/div[3]/form/div[1]/div[1]')
cars.click()
```

```
In [87]: cars_url = []
start = 0
end = 2
for page in range(start, end):
    url = driver.find_elements(By.XPATH, '/html/body/div[1]/div[6]/div[2]/div[1]')
    for i in url:
        fruits_url.append(i.get_attribute('href'))
```

```
In [92]: machine_learning = driver.find_element(By.CLASS_NAME, 'og31Id')
machine_learning.send_keys
```

```
Out[92]: <bound method WebElement.send_keys of <selenium.webdriver.remote.webelement.WebElement (session="a3a5d60132b50b6c5fad664ff59e30a7", element="5CF77996D4E740F8F566C6E91495BDAC_element_6339")>>
```

```
In [93]: machine_learning_url = []
start = 0
end = 2
for page in range(start, end):
    url = driver.find_elements(By.XPATH, '/html/body/div[2]/c-wiz/div[3]/div[1]')
    for i in url:
        machine_learning_url.append(i.get_attribute('href'))
```

```
guitars = driver.find_element(By.XPATH, "/html/body/div[2]/c-wiz/div[3]/div[1],
```

```

-----
WebDriverException                                Traceback (most recent call last)
Cell In[105], line 1
----> 1 guitars = driver.find_element(By.XPATH, "/html/body/div[2]/c-wiz/div
[3]/div[1]/div/div/div/div[1]/c-wiz/div/div")

File ~\anaconda3\Lib\site-packages\selenium\webdriver\remote\webdriver.py:74
2, in WebDriver.find_element(self, by, value)
    739     by = By.CSS_SELECTOR
    740     value = f'[name="{value}"]'
--> 742 return self.execute(Command.FIND_ELEMENT, {"using": by, "value": val
ue})["value"]

File ~\anaconda3\Lib\site-packages\selenium\webdriver\remote\webdriver.py:34
8, in WebDriver.execute(self, driver_command, params)
    346 response = self.command_executor.execute(driver_command, params)
    347 if response:
--> 348     self.error_handler.check_response(response)
    349     response["value"] = self._unwrap_value(response.get("value", Non
e))
    350     return response

File ~\anaconda3\Lib\site-packages\selenium\webdriver\remote\errorhandler.p
y:229, in ErrorHandler.check_response(self, response)
    227     alert_text = value["alert"].get("text")
    228     raise exception_class(message, screen, stacktrace, alert_text)
# type: ignore[call-arg] # mypy is not smart enough here
--> 229 raise exception_class(message, screen, stacktrace)

WebDriverException: Message: disconnected: not connected to DevTools
(failed to check if window was closed: disconnected: not connected to DevT
ools)
(Session info: chrome=121.0.6167.140)
Stacktrace:
    GetHandleVerifier [0x00007FF7EE435E42+3538674]
    (No symbol) [0x00007FF7EE054C02]
    (No symbol) [0x00007FF7EDF05AEB]
    (No symbol) [0x00007FF7EDEF273F]
    (No symbol) [0x00007FF7EDEF22B0]
    (No symbol) [0x00007FF7EDF07BF1]
    (No symbol) [0x00007FF7EDF8B437]
    (No symbol) [0x00007FF7EDF6EE53]
    (No symbol) [0x00007FF7EDF3F514]
    (No symbol) [0x00007FF7EDF40631]
    GetHandleVerifier [0x00007FF7EE466CAD+3738973]
    GetHandleVerifier [0x00007FF7EE4BC506+4089270]
    GetHandleVerifier [0x00007FF7EE4B4823+4057299]
    GetHandleVerifier [0x00007FF7EE185C49+720121]
    (No symbol) [0x00007FF7EE06126F]
    (No symbol) [0x00007FF7EE05C304]
    (No symbol) [0x00007FF7EE05C432]
    (No symbol) [0x00007FF7EE04BD04]
    BaseThreadInitThunk [0x00007FFAD0D9257D+29]
    RtlUserThreadStart [0x00007FFAD198AA58+40]

```

In [106]: 4

Out[106]: 4

```
In [107]: import selenium
import pandas as pd
import time
from selenium import webdriver
from selenium.common.exceptions import StaleElementReferenceException, NoSuchElementException
import requests
from selenium.webdriver.common.by import By
```

In [111]: driver=webdriver.Chrome()

In [112]: driver.get('https://www.flipkart.com')

```
In [148]: Brand_Name=[]
Colour=[]
Storage_RAM_ROM=[]
P_F_Camera=[]
Display_size_Resolution=[]
Processor_And_Cores=[]
Battery=[]
Price=[]
Product_URL=[]
```

```
In [121]: BName=driver.find_elements(By.CLASS_NAME,"Search for Products, Brands and More")
for i in BName:
    if i.text is None :
        Brand_Name.append("--")
    else:
        Brand_Name.append(i.text)
print(len(Brand_Name),Brand_Name)
```

2 ['', '']

In [120]:

Out[120]: []



```
In [126]: ram=driver.find_elements(By.CLASS_NAME,"_1jJQdf _2Mji8F")
for r in ram:
    if i.text is None :
        Storage_RAM_ROM.append("--")
    else:
        Storage_RAM_ROM.append(i.text)
print(len(Storage_RAM_ROM),Storage_RAM_ROM)
```

0 []

In [ ]:

In [ ]:

```
In [135]: PC=driver.find_elements(By.XPATH,"/html/body/div/div/div[3]/div[1]/div[2]/div
for i in PC:
    P_F_Camera.append(i.text)
print(len(P_F_Camera),P_F_Camera)
```

1 ['primary camera']

```
In [137]: DS=driver.find_elements(By.XPATH,"//ul[@class='_1xgFaf']/li[2]")
for i in DS:
    if i.text is None:
        Display_size_Resolution.append("--")
    else:
        Display_size_Resolution.append(i.text)
print(len(Display_size_Resolution),Display_size_Resolution)
```

0 []

```
In [149]: P=driver.find_elements(By.XPATH,"//ul[@class='_1xgFaf']/li[5]")
for i in P:
    if i.text is None :
        Processor_And_Cores.append("--")
    else:
        Processor_And_Cores.append(i.text)
print(len(Processor_And_Cores),Processor_And_Cores)
```

0 []

```
In [150]: B=driver.find_elements(By.XPATH,"//ul[@class='_1xgFaf']//li[4]")
for i in B:
    if i.text is None :
        Battery.append("--")
    else:
        Battery.append(i.text)
print(len(Battery),Battery)
```

0 []

```
In [140]: price=driver.find_elements(By.XPATH,"//div[@class='_30jeq3 _1_WHN1']")
for i in price:
    if i.text is None :
        Price.append("--")
    else:
        Price.append(i.text)
print(len(Price),Price)
```

0 []

```
In [151]: FlipKart=pd.DataFrame(['Brand_Name':Brand_Name,'Storage_RAM_ROM':Storage_RAM_ROM,
```



Cell In[151], line 1

```
FlipKart=pd.DataFrame(['Brand_Name':Brand_Name,'Storage_RAM_ROM':Storage
_RAM_ROM,'P_F_Camera':P_F_Camera,'Display_size_Resolution':Display_size_Reso
lution,'Battery':Battery,'Price':Price])
```

**SyntaxError:** invalid syntax

```
In [152]: 6
```

```
Out[152]: 6
```

```
In [153]: 7
```

```
Out[153]: 7
```

```
In [157]: driver=webdriver.Chrome()
```

```
In [159]: driver.get("https://www.digit.in")
```

```
In [160]: driver.get(url)
```

```
In [161]: Brands=[]  
Products_Description=[]  
Specification=[]  
Price=[]
```

```
In [168]: br=driver.find_elements(By.XPATH,"/html/body/div[1]/div[3]/div/div/article/div
```

```
In [169]: br
```

```
Out[169]: [<selenium.webdriver.remote.webelement.WebElement (session="416d5f5d22905880  
43c49d346060af70", element="8B7E032D26C9DDC757CD175E9AD30F6C_element_19404")  
>]
```

```
In [170]: for i in br:  
  
    Brands.append(i.text)  
Brands
```

```
Out[170]: ['Acer Predator Helios Neo 16 Gaming Laptop']
```

```
In [173]: dp=driver.find_elements(By.XPATH,"/html/body/div[1]/ul/li[1]/a")
```

```
In [174]: dp
```

```
Out[174]: [<selenium.webdriver.remote.webelement.WebElement (session="416d5f5d22905880  
43c49d346060af70", element="3EAA8F1A759A1E3B134FEE1F2E1B2388_element_23250")  
>]
```

```
In [177]: for i in dp:  
  
    Products_Description.append(i.text)  
Products_Description
```

```
Out[177]: ['Description', 'Description', 'Description']
```

```
In [179]: sp=driver.find_elements(By.XPATH,"/html/body/div[1]/ul/li[3]/a")
```

```
In [180]: sp
```

```
Out[180]: [<selenium.webdriver.remote.webelement.WebElement (session="416d5f5d22905880  
43c49d346060af70", element="3EAA8F1A759A1E3B134FEE1F2E1B2388_element_23254")  
>]
```

```
In [181]: for i in sp:
            Specification.append(i.text)
            Specification
```

```
Out[181]: ['Specifications']
```

```
In [182]: len(sp)
```

```
Out[182]: 1
```

```
In [184]: price=driver.find_elements(By.XPATH, "/html/body/div[2]/div[3]/div[1]/div/div/c
```



```
In [185]: price
```

```
Out[185]: [<selenium.webdriver.remote.webelement.WebElement (session="416d5f5d22905880
43c49d346060af70", element="3EAA8F1A759A1E3B134FEE1F2E1B2388_element_21532")
>]
```

```
In [1]: for i in price:
        price.append('i.text')
        price
```

-----  
**NameError** Traceback (most recent call last)

Cell In[1], line 1

```
----> 1 for i in price:
        2     price.append('i.text')
        3 price
```

**NameError**: name 'price' is not defined

```
In [2]: 8
```

```
Out[2]: 8
```

```
In [3]: import selenium
import pandas as pd
import time
from selenium import webdriver
from selenium.common.exceptions import StaleElementReferenceException, NoSuchElementException
import requests
from selenium.webdriver.common.by import By
```

```
In [15]: driver=webdriver.Chrome()
```

```
In [16]: driver.get = ('https://www.youtube.com')
```

```
In [ ]:
```

```
In [ ]:
```

```
In [17]: 9
```

```
Out[17]: 9
```

```
In [1]: import selenium
import pandas as pd
import time
from selenium import webdriver
from selenium.common.exceptions import StaleElementReferenceException, NoSuchElementException
import requests
from selenium.webdriver.common.by import By
```

```
In [ ]:
```

```
In [2]: driver=webdriver.Chrome()
```

```
In [5]: driver.get=("http://www.hostelworld.com")
```

```
In [6]: !pip install bs4
```

```
Requirement already satisfied: bs4 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (0.0.1)
Requirement already satisfied: beautifulsoup4 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from bs4) (4.12.2)
Requirement already satisfied: soupsieve>1.2 in c:\users\ebele okonkwo\anaconda3\lib\site-packages (from beautifulsoup4->bs4) (2.4)
```

```
In [7]: import requests
from bs4 import BeautifulSoup
```

```
In [25]: page = requests.get("https://www.hostelworld.com")
```

```
In [27]: page
```

```
Out[27]: <Response [200]>
```

```
In [26]: url = "https://www.hostelworld.com/hostels/London"
response = requests.get(url)
```

```
In [9]: soup = BeautifulSoup(response.content, "html.parser")
```

```
In [10]: hostels = soup.find_all("div", class_="fabresult")
```

```
In [11]: for hostel in hostels:
    name = hostel.find("h2", class_="fabresult-title").text.strip()
```

```
In [13]: for hostel in hostels:
    distance = hostel.find("span", class_="distance").text.strip()
```

```
In [14]: for hostel in hostels:
    ratings = hostel.find("div", class_="rating").text.strip()
```

```
In [15]: for hostel in hostels:
    total_reviews = hostel.find("div", class_="reviews").text.strip()
```

```
In [16]: for hostel in hostels:
    overall_reviews = hostel.find("div", class_="overall").text.strip()
```

```
In [17]: for hostel in hostels:
    privates_price = hostel.find("div", class_="price-col").find("div", class_="price-col")
```

```
In [18]: for hostel in hostels:
    dorms_price = hostel.find("div", class_="price-col").find("div", class_="price-col")
```



```
In [19]: for hostel in hostels:
    facilities = hostel.find("div", class_="facilities").text.strip()
```

```
In [20]: for hostel in hostels:
    description = hostel.find("div", class_="description").text.strip()
```

```
In [28]:
```

```
In [29]:
```

In [ ]:

In [ ]:

In [ ]: