Task : Web Scrapping From the CSV File

Summary :

Installed the Selenium for the web scrapping and import it to the python file

Defined the path for the chrome driver so it can access the path.

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from csv import DictReader
import json
import time

# defined path for the chromedrivr
PATH="C:\Program Files (x86)\chromedriver.exe"
driver=webdriver.Chrome(PATH)
```

Open the CSV file and iterate and and construct and open the url in browser from the given CSV file and compute the time for every hundred Iteration.

```
 # open the csv file and iterate over the urls contained in
with open('Amazon.csv', 'r') as read_obj:
    csv_dict_reader = DictReader(read_obj)
    count=0
    data={}
    list1=[]
    t0=time.time()
    for row in csv_dict_reader:
        # calculating the time elapsed in processing 100 urls
        if count==100:
                print()
                count=0
                t1=time.time()-t0
                print("Time computed every 100th round of urls : ",t1)
                t0=time.time()

        # Extracting the Country and Asin attributes from csv file
        country=row['country']
        asin=row['Asin']
        # constructing the url to be processed
        url="https://www.amazon."+str(country)+"/dp/"+str(asin)

        # sending url to the driver
        driver.get(url)
```

Defined the try except block for the handling of page not found error . In the inner try except block used for scrapping the product details from the different structures HTML pages. Creating a dictionary to save the details of products and saving them in a list after each iteration

```
    # try and except block to encounter correct url and incorrect url
    try:
            # try except bloack to encounter to extract page data for different types of web pages
            try:
                    # extracting the title , imgurl, price , product details using the id
                    title = driver.find_element(By.ID, "productTitle")
                    imgurl = driver.find_element(By.ID, "imgBlkFront")
                    src=imgurl.get_attribute("src")
                    price = driver.find_element(By.ID, "tmmSwatches")
                    product_details = driver.find_element(By.ID, "detailBulletsWrapper_feature_div")

                    # creating a  dictionary for the details of product
                    product={
                            "Product Title":title.text,
                            "Product Image URL":src,
                            "Price of the Product":price.text,
                            "Product Details":product_details.text,
                            }

                    #appending the dictionary to a list

                    list1.append(product)

            except:
                    # extracting the title , imgurl, price , product details using the id for different stuctured html page
                    title = driver.find_element(By.ID, "productTitle")

                    imgurl = driver.find_element(By.ID, "imgTagWrapperId")
                    src=imgurl.get_attribute("src")
                    price = driver.find_element(By.ID, "tp_price_block_total_price_ww")
                    product_details = driver.find_element(By.ID, "productDescription")

                    # extracting the title , imgurl, price , product details using the id
                    product={
                            "Product Title":title.text,
                            "Product Image URL":src,
                            "Price of the Product":price.text,
```

```
                    # extracting the title , imgurl, price , product details using the id
                    product={
                            "Product Title":title.text,
                            "Product Image URL":src,
                            "Price of the Product":price.text,
                            "Product Details":product_details.text,
                            }
                    #appending the dictionary to a list
                    list1.append(product)


            except:
                    #handling the page not found error and appending to the list
                    urlnotfound={"url_not_available": url}
                    list1.append(urlnotfound)

            count+=1



    # creating the list of dictionaries
    data={'Products':list1}
    # converting dictionary into json
    json_string = json.dumps(data)
    # writing the json file
    with open("ProductsWebScrapping.json", "w") as outfile:
        outfile.write(json_string)
    driver.quit()
```

Creating the list of dictionaries and then converting to json and writing a json file