Group C

Model Implementation

Assignment No. 01

Performance	Understanding	Regularity	Total	Dated Sign of Subject Teacher
03	01	01	05	

Date of Performance: ..... **Date of Completion**: .....

Title: Create a review scrapper for any ecommerce website to fetch real time comments, reviews, ratings, comment tags, customer name using Python.

Objectives:

1. To understand the application and impact of Big Data
2. To understand emerging trends in Big data analytics

Problem Statement:

Create a review scrapper for any E-commerce website to fetch real time comments, reviews, ratings, comment tags, customer name using Python.

Outcomes:

Students will be able to,

1. Apply the Analytical concept of Big data using Python.
2. Design and develop Big data analytic application for emerging trends

Software and Hardware requirements:

3. Software: Ubuntu OS/Windows, Anaconda, Jupyter notebook
4. Hardware: Processor, Ethernet Connection or WiFi, RAM 1GB, HDD, Sound Card, camera, microphone (depending upon website selection)

Theory:

In this practical, we are going to see how we can scrape the E-commerce Website like Amazon customer review using Beautiful Soup in Python.

Module Needed:

bs4: Beautiful Soup (bs4) is a Python library for pulling data out of HTML and XML files. This module does not come built-in with Python. To install this type the below command in the terminal.

```
pip install bs4
```

requests: Request allows you to send HTTP/1.1 requests extremely easily. This module also does not come built-in with Python. To install this type the below command in the terminal.

```
pip install requests
```

To begin with web scraping, we first have to do some setup. Import all the required modules. Get the cookies data for making the request to amazon, without this you can not able to scrape. Create a header that contains your request cookies, without cookies you can not scrape amazon data it always shows some error.

Pass the URL in the getdata() function(User Defined Function) to that will request to a URL, it returns a response. We are using get method to retrieve information from the given server using a given URL.

Syntax: `requests.get(url, args)`

Convert that data into HTML code and then Parse the HTML content using bs4.

Syntax: `soup = BeautifulSoup(r.content, 'html5lib')`

Parameters:

- **r.content** : It is the raw HTML content.
- **html.parser** : Specifying the HTML parser we want to use.

Scrape customer name:

Now find the customer list with span tag where **class_ = a-profile-name**. You can open the webpage in the browser and inspect the relevant element by pressing right-click as shown in the figure.

You have to pass the tag name and attribute with its corresponding value to the **find_all()** function.

```
def cus_data(soup):
    # find the Html tag
    # with find()
    # and convert into string
    data_str = ""
    cus_list = []

    for item in soup.find_all("span", class_="a-profile-name"):
        data_str = data_str + item.get_text()
        cus_list.append(data_str)
        data_str = ""
    return cus_list


cus_res = cus_data(soup)
print(cus_res)
```

Output:

['Amaze', 'Robert', 'D. Kong', 'Alexey', 'Charl', 'RBostillo']

Scrape user review:

Now find the customer review as same above methods. Find the unique class name with a specific tag, here we use div tag.

```
def cus_rev(soup):
        # find the Html tag
        # with find()
        # and convert into string
        data_str = ""

        for item in soup.find_all("div", class_="a-expander-content
    \
        reviewText review-text-content a-expander-partial-
    collapse-content"):
            data_str = data_str + item.get_text()

        result = data_str.split("\n")
        return (result)



    rev_data = cus_rev(soup)
    rev_result = []
    for i in rev_data:
        if i is "":
            pass
        else:
            rev_result.append(i)
    rev_result
```

Scraping Production information

Here we will scrape product information like product name, ASIN number, Weight, dimension. By doing this we will use the span tag and with a specific unique class name.

```
        def product_info(soup):

            # find the Html tag
            # with find()
            # and convert into string
            data_str = ""
            pro_info = []

            for item in soup.find_all("ul", class_="a-unordered-
        list a-nostyle\
            a-vertical a-spacing-none detail-bullet-list"):
```

```
            data_str = data_str + item.get_text()
            pro_info.append(data_str.split("\n"))
            data_str = ""
        return pro_info


pro_result = product_info(soup)

# Filter the required data
for item in pro_result:
    for j in item:
        if j is "":
            pass
        else:
            print(j)
```

Scraping Review Image:

Here we will extract the image link from the review of the product using the same as the above methods. The tag name and attribute of the tag is passed to **findAll()** as above.

```
def rev_img(soup):

    # find the Html tag
    # with find()
    # and convert into string
    data_str = ""
    cus_list = []
    images = []
    for img in soup.findAll('img', class_="cr-lightbox-
image-thumbnail"):
        images.append(img.get('src'))
    return images


img_result = rev_img(soup)
img_result
```

Saving details into CSV file:

Here we will save the details into the CSV file, We will convert the data into dataframe and then export it into the CSV, Let us see how to export a Pandas DataFrame to a CSV file. We will be using the to_csv() function to save a DataFrame as a CSV file.

Syntax : `to_csv(parameters)`
Parameters :

- **path_or_buf :** File path or object, if None is provided the result is returned as a string.

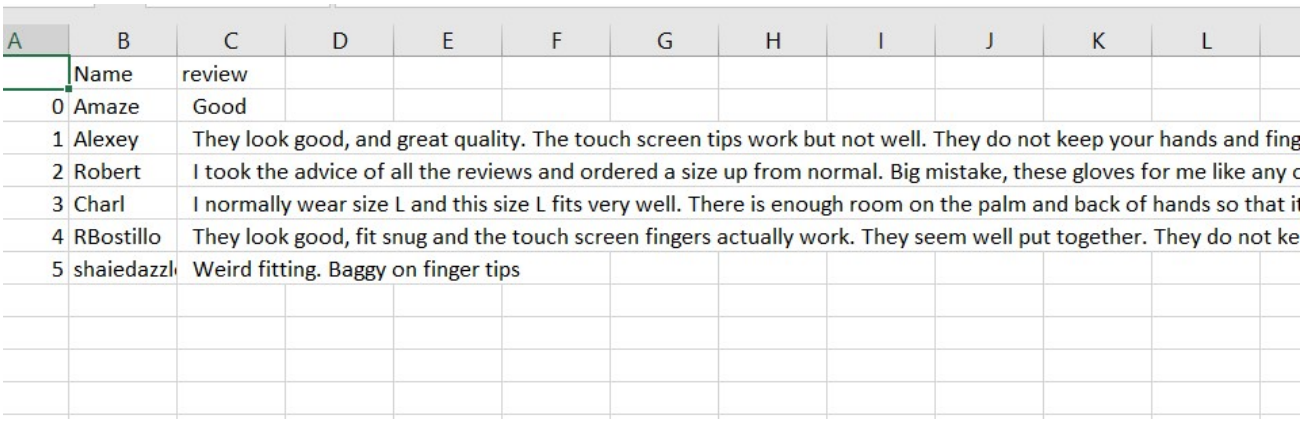
```
import pandas as pd

# initialise data of lists.
data = {'Name': cus_res,
        'review': rev_result}

# Create DataFrame
df = pd.DataFrame(data)

# Save the output.
df.to_csv('amazon_review.csv')
```

Output:



	Name	review
0	Amaze	Good
1	Alexey	They look good, and great quality. The touch screen tips work but not well. They do not keep your hands and fing
2	Robert	I took the advice of all the reviews and ordered a size up from normal. Big mistake, these gloves for me like any c
3	Charl	I normally wear size L and this size L fits very well. There is enough room on the palm and back of hands so that it
4	RBostillo	They look good, fit snug and the touch screen fingers actually work. They seem well put together. They do not ke
5	shaiedazzl	Weird fitting. Baggy on finger tips

Conclusion:

Finally we have managed to scrape all the information from every page of E-commerce website.

FAQ:

1) What are things to remember while scraping Amazon Product Reviews?
2) Why to Scrape Amazon Product Reviews?
3) Which modules are needed for scrape the E-commerce Website like Amazon customer review?