Vidyavardhaka College of Engineering,

Autonomous Institute, Affiliated to Visveswaraya Technological University, Belagavi.
Accredited by NBA, New Delhi & NAAC with 'A' Grade
Gokulam 3rd Stage, Mysuru - 570002, Karnataka, India

Advanced Python Laboratory (BISAB317)

Activity Based Assessment

Report on

"WEB SCRAPING"

of customer reviews

Submitted by,
Tejaswini R
Punya V Gowda

Submitted to,
Prof.Kasturi Rangan
Department of ISE, VVCE

Introduction of Web Scraping

Web scraping is a data extraction technique employed to gather information from websites automatically. Through the use of specialized software or programming scripts, users can navigate the structure of web pages, locate specific data elements, and extract them for further analysis. This process is invaluable for various applications, including market research, competitive analysis, and real-time data monitoring. Web scraping plays a crucial role in aggregating data from diverse sources, aiding businesses and researchers in making informed decisions. However, it's essential to approach web scraping ethically and responsibly, respecting the terms of service of the websites being scraped and avoiding any undue strain on server resources. Despite its utility, web scraping is a practice that requires careful consideration of legal and ethical implications. Users must be aware of and comply with relevant laws and regulations to ensure they are not infringing on copyrights or violating privacy policies. Responsible web scraping practices contribute to maintaining a balance between data accessibility and the protection of online content and user privacy.

Motivation

Web scraping is the automated process of extracting data from websites. It involves using specialized tools or programming to navigate through web pages, locate specific information, and retrieve it for analysis.

Unlocking Possibilities: Combining web scraping with chat bots opens up a world of opportunities in today's fast-paced digital era, where information is key.

Simplifying Data Acquisition: The main reason for integrating web scraping into chatbots is to make data gathering easier. Users no longer need coding skills or complex tools; they can simply tell the chatbot what they need, and it does the rest.

PROBLEM STATEMENT

The steps involved in web scraping typically include:

1. Identifying Target Website: Determine the website from which you want to extract data.
2. Inspecting the Website Structure: Understand the HTML structure of the target website to locate the data elements you need.
3. Selecting a Scraping Tool or Library: Choose a suitable tool or programming library such as BeautifulSoup or Scrapy in Python to facilitate the scraping process.
4. Sending HTTP Requests: Use the scraping tool to send HTTP requests to the website's server and retrieve the HTML content of the pages

. 5. Parsing HTML Content: Parse the HTML content to extract the relevant data by identifying tags, classes, or other HTML elements

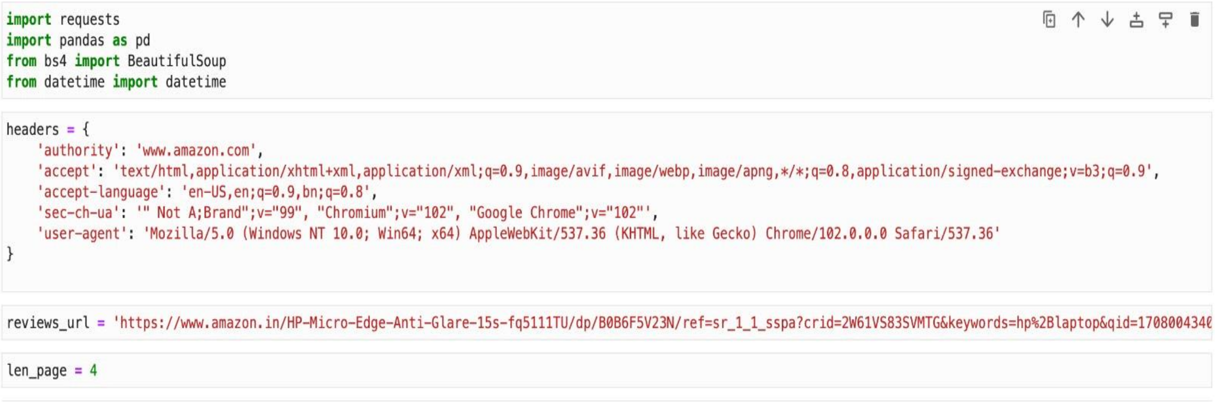
. 6. Storing the Extracted Data: Save the extracted data in a structured format like CSV, JSON, or a database for further analysis.

7. Handling Pagination and Navigation: If the data spans multiple pages, implement logic to navigate through pagination and extract data from each page

. 8. Respecting Robots.txt and Terms of Service: Adhere to ethical guidelines by checking the website's robots.txt file and respecting its terms of service to ensure compliance with legal and ethical standards.

DESIGN & IMPLEMENTATION OF CODE

1. Making a Request



```
import requests
import pandas as pd
from bs4 import BeautifulSoup
from datetime import datetime
```

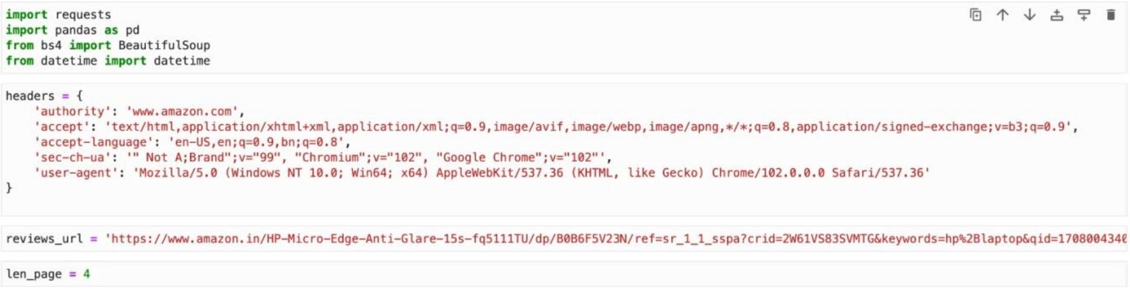
```
headers = {
    'authority': 'www.amazon.com',
    'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9',
    'accept-language': 'en-US,en;q=0.9,bn;q=0.8',
    'sec-ch-ua': '" Not A;Brand";v="99", "Chromium";v="102", "Google Chrome";v="102"',
    'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.0.0 Safari/537.36'
}
```

```
reviews_url = 'https://www.amazon.in/HP-Micro-Edge-Anti-Glare-15s-fq5111TU/dp/B0B6F5V23N/ref=sr_1_1_sspa?crid=2W61VS83SVMTG&keywords=hp%2Blaptop&qid=1708004340
```

```
len_page = 4
```

In simple terms, this code uses Python to fetch the content of a webpage and print it out. It shows the response status code to confirm if the request was successful, and then prints the webpage's content, which includes its HTML code.

2.Extracting URLs



```
import requests
import pandas as pd
from bs4 import BeautifulSoup
from datetime import datetime
```

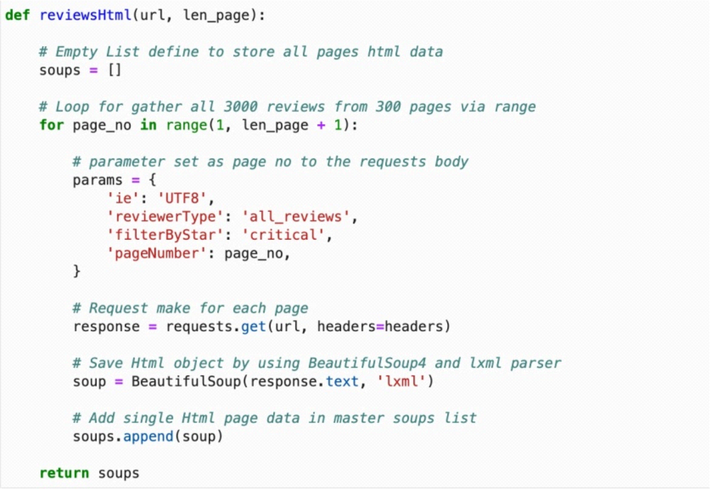
```
headers = {
    'authority': 'www.amazon.com',
    'accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9',
    'accept-language': 'en-US,en;q=0.9,bn;q=0.8',
    'sec-ch-ua': '" Not A;Brand";v="99", "Chromium";v="102", "Google Chrome";v="102"',
    'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.0.0 Safari/537.36'
}
```

```
reviews_url = 'https://www.amazon.in/HP-Micro-Edge-Anti-Glare-15s-fq5111TU/dp/B0B6F5V23N/ref=sr_1_1_sspa?crid=2W61VS83SVMTG&keywords=hp%2Blaptop&qid=1708004340
```

```
len_page = 4
```

This code sends a request to a webpage, then prints the URL of the request and the status code of the response.

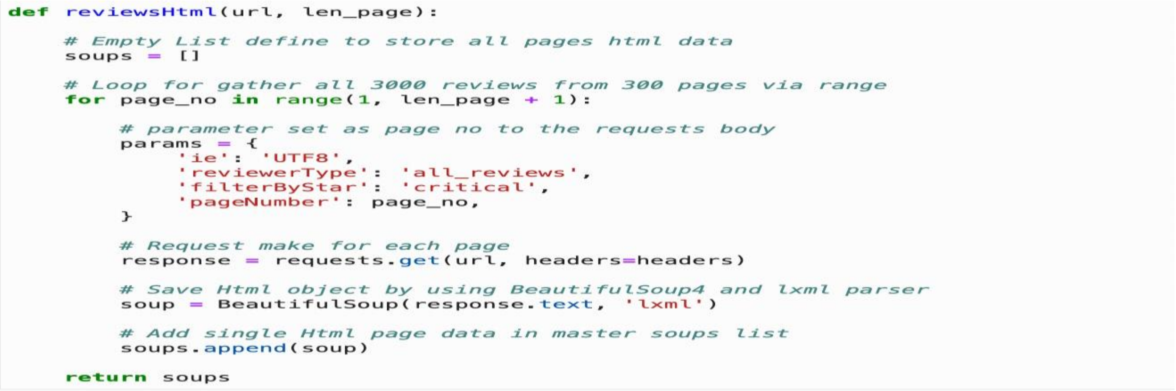
3.Installing a Beautiful Soup Library



```
def reviewsHtml(url, len_page):

    # Empty List define to store all pages html data
    soups = []

    # Loop for gather all 3000 reviews from 300 pages via range
    for page_no in range(1, len_page + 1):

        # parameter set as page no to the requests body
        params = {
            'ie': 'UTF8',
            'reviewerType': 'all_reviews',
            'filterByStar': 'critical',
            'pageNumber': page_no,
        }

        # Request make for each page
        response = requests.get(url, headers=headers)

        # Save Html object by using BeautifulSoup4 and lxml parser
        soup = BeautifulSoup(response.text, 'lxml')

        # Add single Html page data in master soups list
        soups.append(soup)

    return soups
```

This code fetches a webpage, prints its status code, then uses BeautifulSoup to parse the HTML content and prints it in a prettified format

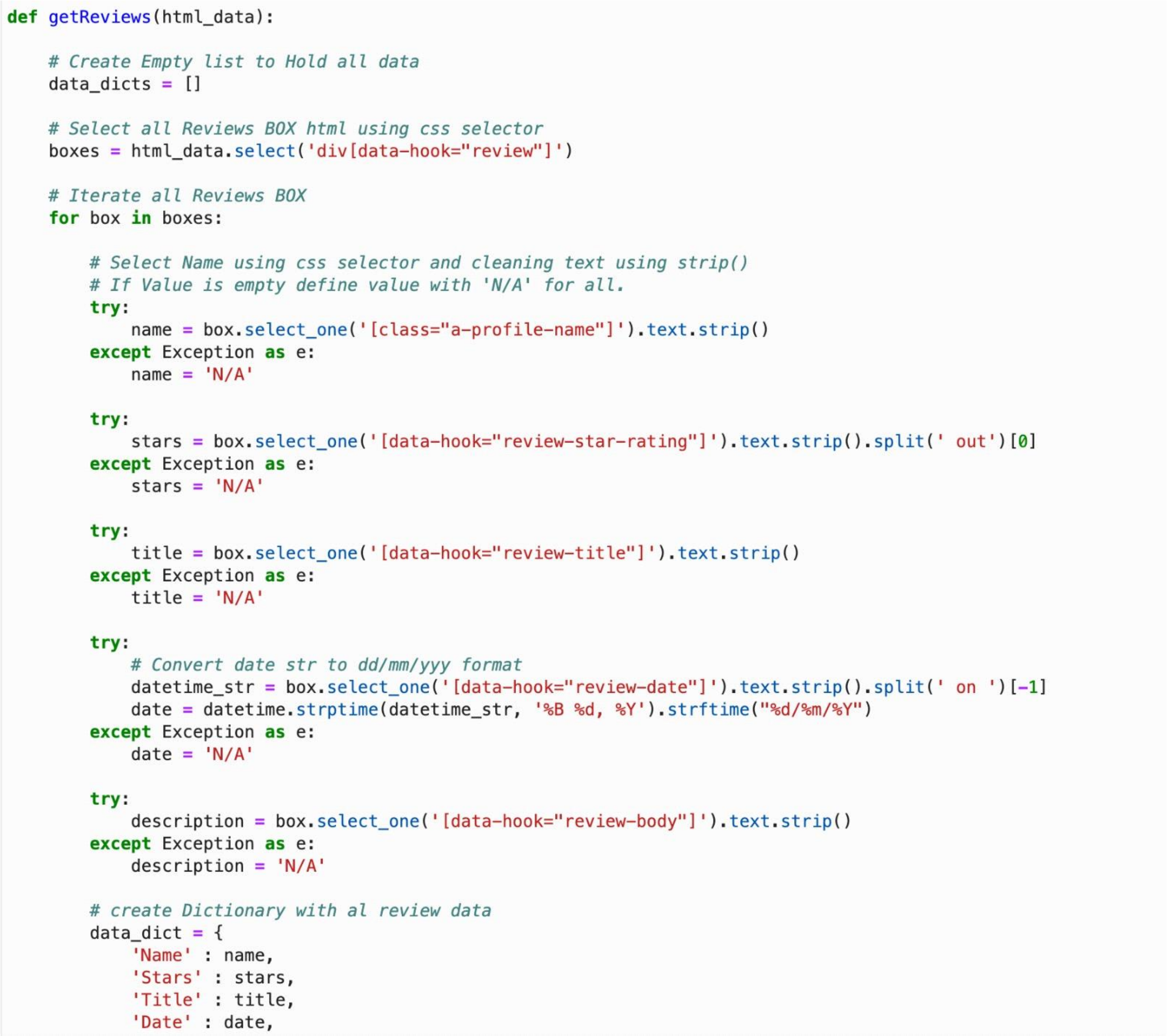
4.Parsing using Beautiful Soup



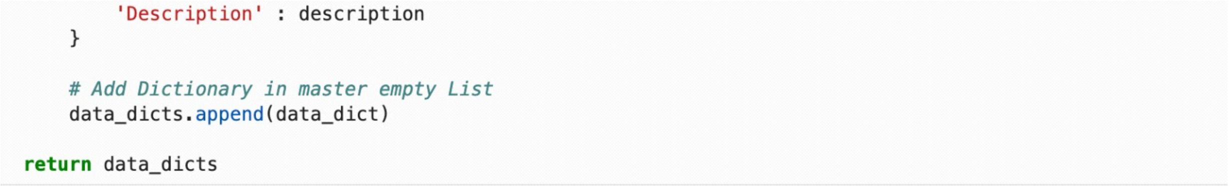
```
def reviewsHtml(url, len_page):

    # Empty List define to store all pages html data
    soups = []

    # Loop for gather all 3000 reviews from 300 pages via range
    for page_no in range(1, len_page + 1):

        # parameter set as page no to the requests body
        params = {
            'ie': 'UTF8',
            'reviewerType': 'all_reviews',
            'filterByStar': 'critical',
            'pageNumber': page_no,
        }

        # Request make for each page
        response = requests.get(url, headers=headers)

        # Save Html object by using BeautifulSoup4 and lxml parser
        soup = BeautifulSoup(response.text, 'lxml')

        # Add single Html page data in master soups list
        soups.append(soup)

    return soups
```

This code fetches a webpage, parses its HTML content using BeautifulSoup, then extracts and prints information about the title tag, its name, parent tag name, and the name of its child tag.

5. Extract review data from HTML content.



```python
def getReviews(html_data):

    # Create Empty list to Hold all data
    data_dicts = []

    # Select all Reviews BOX html using css selector
    boxes = html_data.select('div[data-hook="review"]')

    # Iterate all Reviews BOX
    for box in boxes:

        # Select Name using css selector and cleaning text using strip()
        # If Value is empty define value with 'N/A' for all.
        try:
            name = box.select_one('[class="a-profile-name"]').text.strip()
        except Exception as e:
            name = 'N/A'

        try:
            stars = box.select_one('[data-hook="review-star-rating"]').text.strip().split(' out')[0]
        except Exception as e:
            stars = 'N/A'

        try:
            title = box.select_one('[data-hook="review-title"]').text.strip()
        except Exception as e:
            title = 'N/A'

        try:
            # Convert date str to dd/mm/yyy format
            datetime_str = box.select_one('[data-hook="review-date"]').text.strip().split(' on ')[-1]
            date = datetime.strptime(datetime_str, '%B %d, %Y').strftime("%d/%m/%Y")
        except Exception as e:
            date = 'N/A'

        try:
            description = box.select_one('[data-hook="review-body"]').text.strip()
        except Exception as e:
            description = 'N/A'

        # create Dictionary with al review data
        data_dict = {
            'Name' : name,
            'Stars' : stars,
            'Title' : title,
            'Date' : date,
```



```python
        'Description' : description
    }

    # Add Dictionary in master empty List
    data_dicts.append(data_dict)

return data_dicts
```

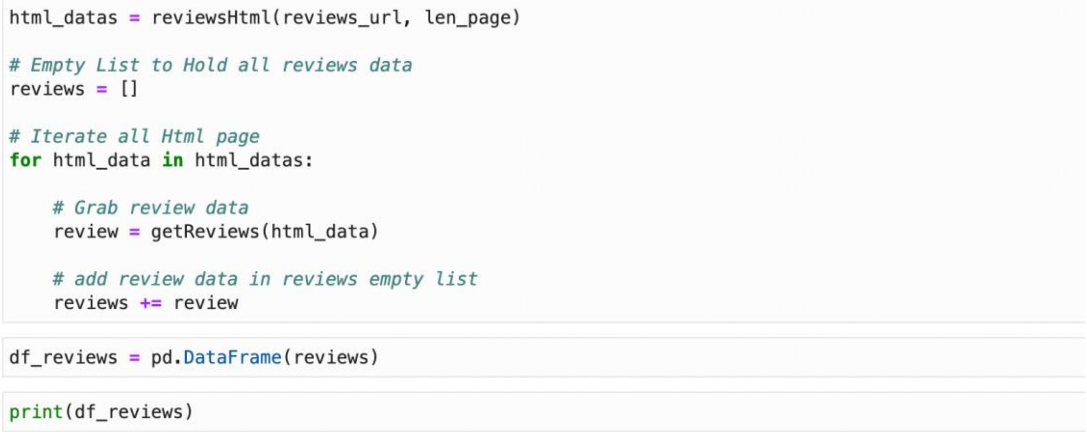
Input: The function takes HTML data as input.

Processing:
It initializes an empty list called data_dicts to store all the review data dictionaries.
It selects review elements from the HTML using CSS selectors.
It iterates over each review element and extracts relevant information like reviewer name, stars, title, date, and description. If any of these elements are missing or encounter an error during extraction, it handles it gracefully by setting a default value ('N/A').

It converts the date string from one format to another using datetime.strptime and strftime methods to ensure consistency.
It creates a dictionary (data_dict) for each review with the extracted information and appends it to the list data_dicts.

Output: The function returns a list of dictionaries containing the extracted review data.

6. Retrieves review data from multiple HTML pages and accumulates them into a list.



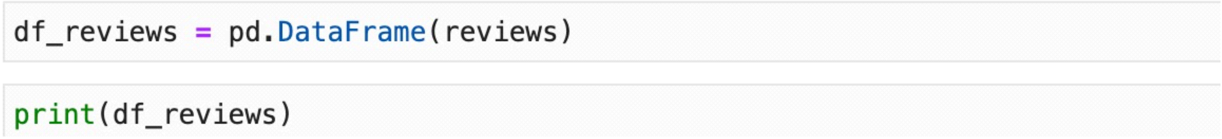
```
html_datas = reviewsHtml(reviews_url, len_page)

# Empty List to Hold all reviews data
reviews = []

# Iterate all Html page
for html_data in html_datas:

    # Grab review data
    review = getReviews(html_data)

    # add review data in reviews empty list
    reviews += review
```

```
df_reviews = pd.DataFrame(reviews)
```

```
print(df_reviews)
```

html_datas = reviewsHtml(reviews_url, len_page): This line presumably retrieves HTML data for reviews from multiple pages using the reviewsHtml function, based on the provided URL (reviews_url) and the number of pages (len_page).
reviews = []: This initializes an empty list called reviews to store all the review data.
Iteration over HTML pages:
The code iterates through each HTML data obtained from html_datas.
For each HTML page, it extracts review data using the getReviews function.
The obtained review data (review) is appended to the reviews list using the += operator, which concatenates the lists.
Output: At the end of this process, the reviews list will contain review data extracted from all the HTML pages.

7. Creates a Pandas DataFrame and Prints

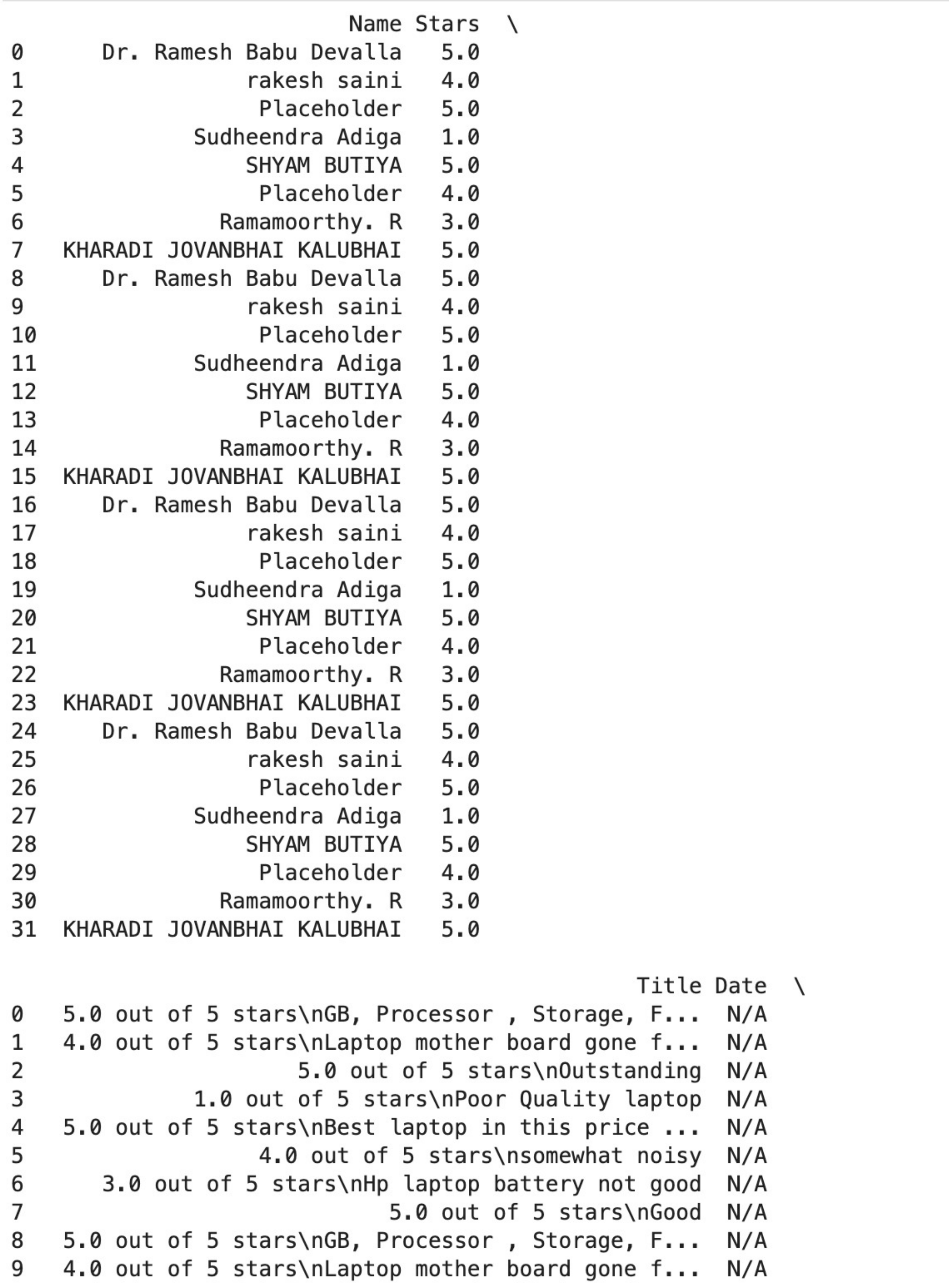


```
df_reviews = pd.DataFrame(reviews)
```

```
print(df_reviews)
```

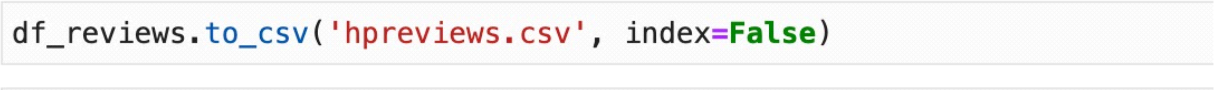
pd.DataFrame(reviews): This function call converts the list of dictionaries (reviews) into a DataFrame object. Each dictionary within the list represents a row in the DataFrame, where the keys of the dictionaries become column names, and the corresponding values become the data in the DataFrame.

Output: df_reviews will be a structured tabular data format where each row represents a review, and columns correspond to different attributes of the reviews such as reviewer name, stars, title, date, and description.
And prints the output.



```
                            Name Stars  \
0       Dr. Ramesh Babu Devalla   5.0
1                  rakesh saini   4.0
2                   Placeholder   5.0
3              Sudheendra Adiga   1.0
4                   SHYAM BUTIYA   5.0
5                   Placeholder   4.0
6                Ramamoorthy. R   3.0
7   KHARADI JOVANBHAI KALUBHAI   5.0
8       Dr. Ramesh Babu Devalla   5.0
9                  rakesh saini   4.0
10                  Placeholder   5.0
11             Sudheendra Adiga   1.0
12                 SHYAM BUTIYA   5.0
13                  Placeholder   4.0
14               Ramamoorthy. R   3.0
15  KHARADI JOVANBHAI KALUBHAI   5.0
16      Dr. Ramesh Babu Devalla   5.0
17                 rakesh saini   4.0
18                  Placeholder   5.0
19             Sudheendra Adiga   1.0
20                 SHYAM BUTIYA   5.0
21                  Placeholder   4.0
22               Ramamoorthy. R   3.0
23  KHARADI JOVANBHAI KALUBHAI   5.0
24      Dr. Ramesh Babu Devalla   5.0
25                 rakesh saini   4.0
26                  Placeholder   5.0
27             Sudheendra Adiga   1.0
28                 SHYAM BUTIYA   5.0
29                  Placeholder   4.0
30               Ramamoorthy. R   3.0
31  KHARADI JOVANBHAI KALUBHAI   5.0

                                                Title Date  \
0   5.0 out of 5 stars\nGB, Processor , Storage, F...  N/A
1   4.0 out of 5 stars\nLaptop mother board gone f...  N/A
2                    5.0 out of 5 stars\nOutstanding  N/A
3             1.0 out of 5 stars\nPoor Quality laptop  N/A
4   5.0 out of 5 stars\nBest laptop in this price ...  N/A
5                 4.0 out of 5 stars\nsomewhat noisy  N/A
6       3.0 out of 5 stars\nHp laptop battery not good  N/A
7                           5.0 out of 5 stars\nGood  N/A
8   5.0 out of 5 stars\nGB, Processor , Storage, F...  N/A
9   4.0 out of 5 stars\nLaptop mother board gone f...  N/A
```

8.Create a CSV file

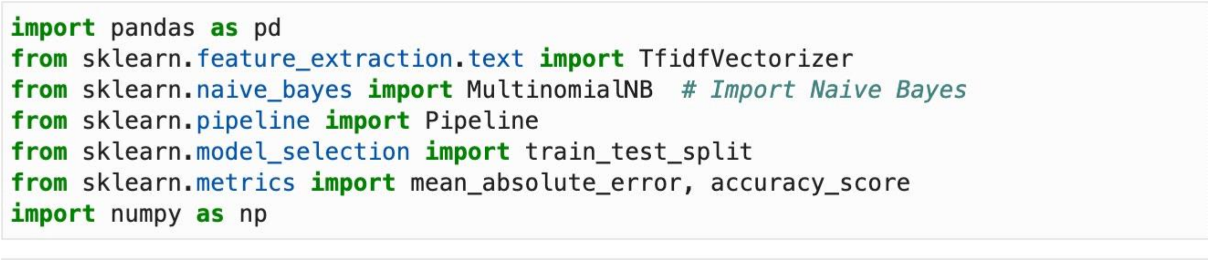


```
df_reviews.to_csv('hpreviews.csv', index=False)
```

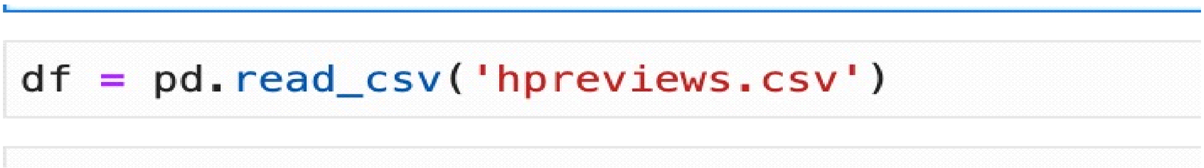
df_reviews.to_csv('hpreviews.csv', index=False): This method call writes the contents of the DataFrame df_reviews to a CSV file named 'hpreviews.csv'. The parameter index=False specifies that the DataFrame index should not be included in the exported file.

Output: After execution, a CSV file named 'hpreviews.csv' will be created in the current directory containing the review data from the DataFrame df_reviews, with each row representing a review and each column representing different attributes of the reviews.

9.Imports several modules from the scikit-learn library (also known as sklearn) and NumPy.

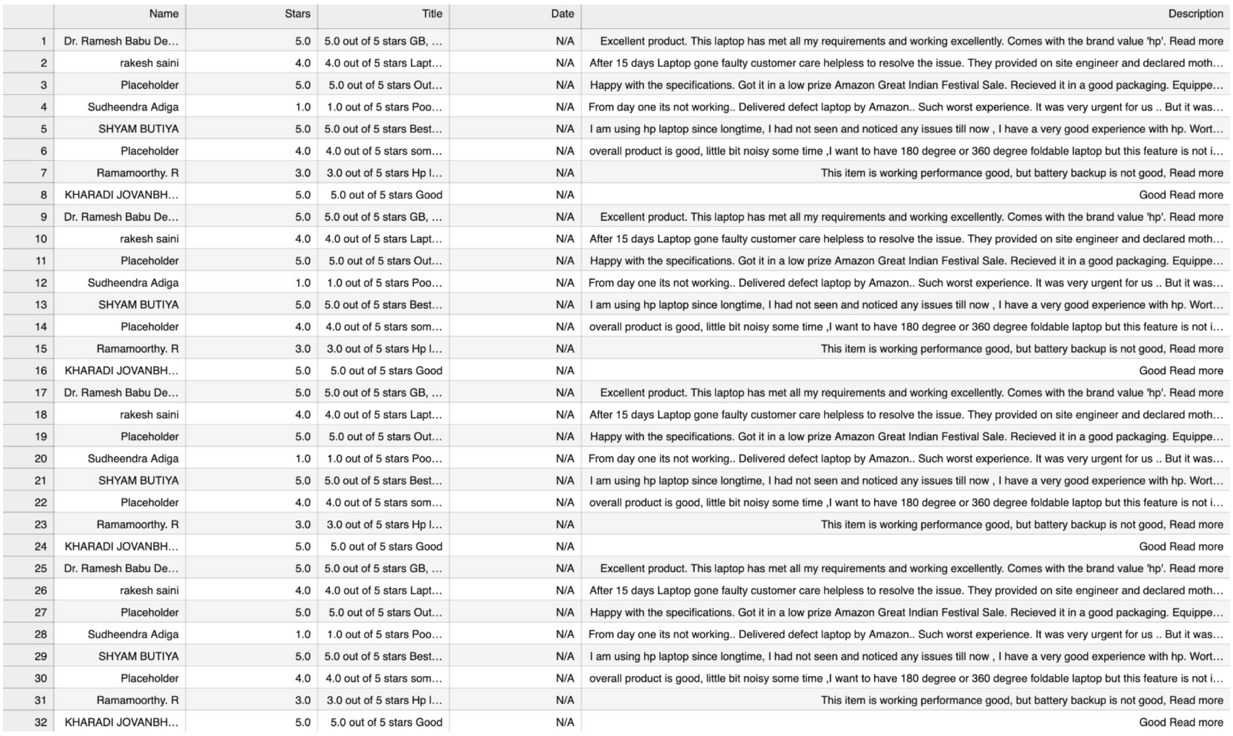


```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.naive_bayes import MultinomialNB  # Import Naive Bayes
from sklearn.pipeline import Pipeline
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_absolute_error, accuracy_score
import numpy as np
```



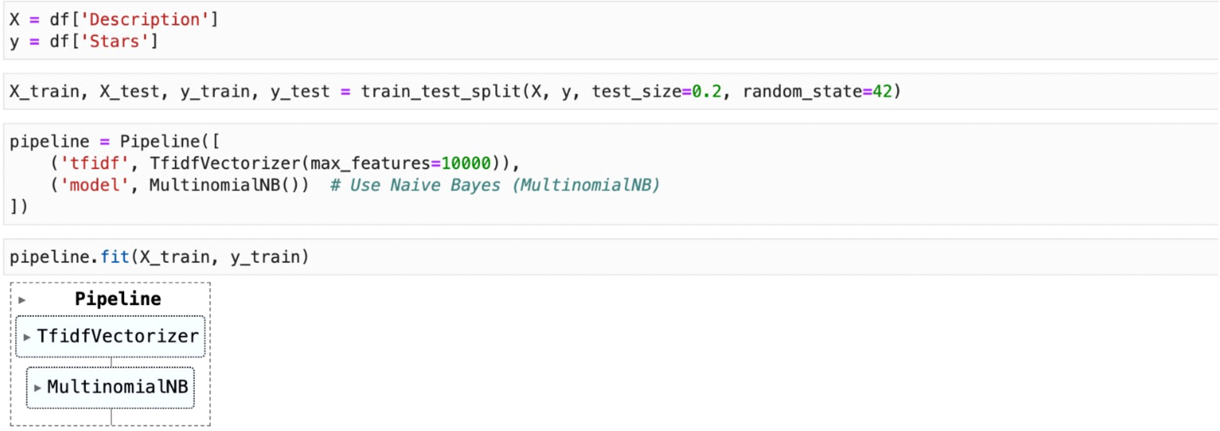
```
df = pd.read_csv('hpreviews.csv')
```

The code imports essential libraries for machine learning tasks in Python. Pandas handles data structures, scikit-learn offers tools for machine learning algorithms and evaluation. NumPy supports numerical operations. Specifically, it imports tools for text vectorization, Naive Bayes classification, pipeline creation, data splitting, and evaluation metrics.
Reads a CSV file.



	Name	Stars	Title	Date	Description
1	Dr. Ramesh Babu De…	5.0	5.0 out of 5 stars GB, …	N/A	Excellent product. This laptop has met all my requirements and working excellently. Comes with the brand value 'hp'. Read more
2	rakesh saini	4.0	4.0 out of 5 stars Lapt…	N/A	After 15 days Laptop gone faulty customer care helpless to resolve the issue. They provided on site engineer and declared moth…
3	Placeholder	5.0	5.0 out of 5 stars Out…	N/A	Happy with the specifications. Got it in a low prize Amazon Great Indian Festival Sale. Recieved it in a good packaging. Equippe…
4	Sudheendra Adiga	1.0	1.0 out of 5 stars Poo…	N/A	From day one its not working.. Delivered defect laptop by Amazon.. Such worst experience. It was very urgent for us .. But it was…
5	SHYAM BUTIYA	5.0	5.0 out of 5 stars Best…	N/A	I am using hp laptop since longtime, I had not seen and noticed any issues till now , I have a very good experience with hp. Wort…
6	Placeholder	4.0	4.0 out of 5 stars som…	N/A	overall product is good, little bit noisy some time ,I want to have 180 degree or 360 degree foldable laptop but this feature is not i…
7	Ramamoorthy. R	3.0	3.0 out of 5 stars Hp l…	N/A	This item is working performance good, but battery backup is not good, Read more
8	KHARADI JOVANBH…	5.0	5.0 out of 5 stars Good	N/A	Good Read more
9	Dr. Ramesh Babu De…	5.0	5.0 out of 5 stars GB, …	N/A	Excellent product. This laptop has met all my requirements and working excellently. Comes with the brand value 'hp'. Read more
10	rakesh saini	4.0	4.0 out of 5 stars Lapt…	N/A	After 15 days Laptop gone faulty customer care helpless to resolve the issue. They provided on site engineer and declared moth…
11	Placeholder	5.0	5.0 out of 5 stars Out…	N/A	Happy with the specifications. Got it in a low prize Amazon Great Indian Festival Sale. Recieved it in a good packaging. Equippe…
12	Sudheendra Adiga	1.0	1.0 out of 5 stars Poo…	N/A	From day one its not working.. Delivered defect laptop by Amazon.. Such worst experience. It was very urgent for us .. But it was…
13	SHYAM BUTIYA	5.0	5.0 out of 5 stars Best…	N/A	I am using hp laptop since longtime, I had not seen and noticed any issues till now , I have a very good experience with hp. Wort…
14	Placeholder	4.0	4.0 out of 5 stars som…	N/A	overall product is good, little bit noisy some time ,I want to have 180 degree or 360 degree foldable laptop but this feature is not i…
15	Ramamoorthy. R	3.0	3.0 out of 5 stars Hp l…	N/A	This item is working performance good, but battery backup is not good, Read more
16	KHARADI JOVANBH…	5.0	5.0 out of 5 stars Good	N/A	Good Read more
17	Dr. Ramesh Babu De…	5.0	5.0 out of 5 stars GB, …	N/A	Excellent product. This laptop has met all my requirements and working excellently. Comes with the brand value 'hp'. Read more
18	rakesh saini	4.0	4.0 out of 5 stars Lapt…	N/A	After 15 days Laptop gone faulty customer care helpless to resolve the issue. They provided on site engineer and declared moth…
19	Placeholder	5.0	5.0 out of 5 stars Out…	N/A	Happy with the specifications. Got it in a low prize Amazon Great Indian Festival Sale. Recieved it in a good packaging. Equippe…
20	Sudheendra Adiga	1.0	1.0 out of 5 stars Poo…	N/A	From day one its not working.. Delivered defect laptop by Amazon.. Such worst experience. It was very urgent for us .. But it was…
21	SHYAM BUTIYA	5.0	5.0 out of 5 stars Best…	N/A	I am using hp laptop since longtime, I had not seen and noticed any issues till now , I have a very good experience with hp. Wort…
22	Placeholder	4.0	4.0 out of 5 stars som…	N/A	overall product is good, little bit noisy some time ,I want to have 180 degree or 360 degree foldable laptop but this feature is not i…
23	Ramamoorthy. R	3.0	3.0 out of 5 stars Hp l…	N/A	This item is working performance good, but battery backup is not good, Read more
24	KHARADI JOVANBH…	5.0	5.0 out of 5 stars Good	N/A	Good Read more
25	Dr. Ramesh Babu De…	5.0	5.0 out of 5 stars GB, …	N/A	Excellent product. This laptop has met all my requirements and working excellently. Comes with the brand value 'hp'. Read more
26	rakesh saini	4.0	4.0 out of 5 stars Lapt…	N/A	After 15 days Laptop gone faulty customer care helpless to resolve the issue. They provided on site engineer and declared moth…
27	Placeholder	5.0	5.0 out of 5 stars Out…	N/A	Happy with the specifications. Got it in a low prize Amazon Great Indian Festival Sale. Recieved it in a good packaging. Equippe…
28	Sudheendra Adiga	1.0	1.0 out of 5 stars Poo…	N/A	From day one its not working.. Delivered defect laptop by Amazon.. Such worst experience. It was very urgent for us .. But it was…
29	SHYAM BUTIYA	5.0	5.0 out of 5 stars Best…	N/A	I am using hp laptop since longtime, I had not seen and noticed any issues till now , I have a very good experience with hp. Wort…
30	Placeholder	4.0	4.0 out of 5 stars som…	N/A	overall product is good, little bit noisy some time ,I want to have 180 degree or 360 degree foldable laptop but this feature is not i…
31	Ramamoorthy. R	3.0	3.0 out of 5 stars Hp l…	N/A	This item is working performance good, but battery backup is not good, Read more
32	KHARADI JOVANBH…	5.0	5.0 out of 5 stars Good	N/A	Good Read more

10.Splitting into training and testing sets using Multinomial Naïve Bayes



```
X = df['Description']
y = df['Stars']
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

```
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(max_features=10000)),
    ('model', MultinomialNB())  # Use Naive Bayes (MultinomialNB)
])
```

```
pipeline.fit(X_train, y_train)
```

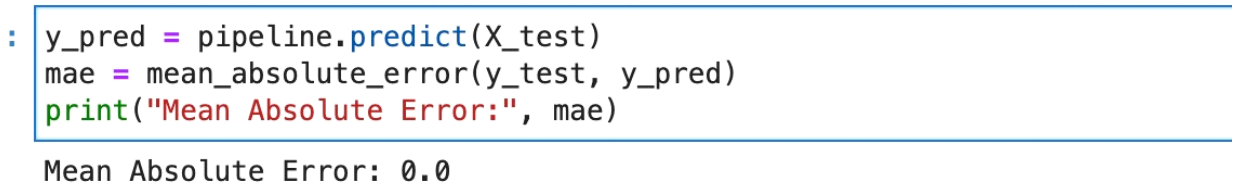
Data Preparation:
X = df['Description'] extracts the 'Description' column from the DataFrame df as the feature (input).
y = df['Stars'] extracts the 'Stars' column from the DataFrame df as the target (output).

Data Splitting:
train_test_split(X, y, test_size=0.2, random_state=42) splits the data into training and testing sets. 20% of the data is allocated for testing while the remaining 80% is for training. The random_state=42 ensures reproducibility.

Pipeline Definition:
Pipeline chains together multiple processing steps.
TfidfVectorizer(max_features=10000) converts text data into TF-IDF (Term Frequency-Inverse Document Frequency) features, limiting the number of features to 10,000 to reduce complexity.
MultinomialNB() initializes the Multinomial Naive Bayes classifier, a common choice for text classification tasks.

Model Training:
pipeline.fit(X_train, y_train) trains the pipeline on the training data, where X_train is the input features and y_train is the target variable.

11.Finding Mean Absolute Error

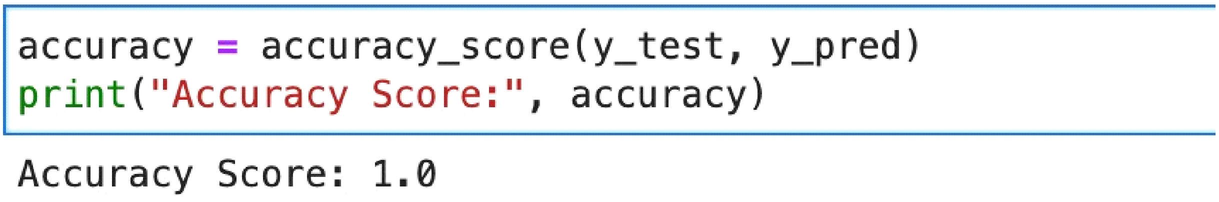


```
y_pred = pipeline.predict(X_test)
mae = mean_absolute_error(y_test, y_pred)
print("Mean Absolute Error:", mae)
```

```
Mean Absolute Error: 0.0
```

This code segment predicts star ratings (y_pred) for the test data (X_test) using the trained pipeline. It then calculates the mean absolute error (mae) between the predicted ratings and the actual ratings (y_test). Finally, it prints the mean absolute error as a measure of the model's performance in predicting star ratings accurately.

12.Finding Accuracy Score

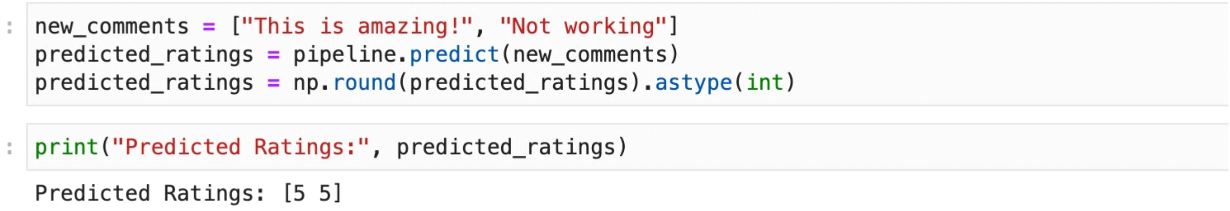


```
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy Score:", accuracy)
```

```
Accuracy Score: 1.0
```

The code calculates the accuracy score (accuracy) by comparing the predicted star ratings (y_pred) with the actual ratings (y_test). It measures the proportion of correct predictions out of all predictions made on the test dataset. Finally, it prints the accuracy score, indicating the model's overall predictive accuracy.

13.Finding Predicted Ratings



```
new_comments = ["This is amazing!", "Not working"]
predicted_ratings = pipeline.predict(new_comments)
predicted_ratings = np.round(predicted_ratings).astype(int)
```

```
print("Predicted Ratings:", predicted_ratings)
```

```
Predicted Ratings: [5 5]
```

The code predicts star ratings for new comments provided in the list new_comments using the trained pipeline (pipeline). It then rounds the predicted ratings to the nearest integer and converts them to integers. Finally, it prints the predicted ratings for the new comments as an array.

CONCLUSION

In conclusion, web scraping is a powerful tool for automating the extraction of valuable data from websites, facilitating tasks ranging from market research to data analysis. While offering immense benefits, it is crucial to approach web scraping responsibly, respecting legal and ethical considerations. Adherence to website terms of service, respect for privacy, and consideration of server load are essential elements in maintaining a balance between data accessibility and ethical practices. When employed judiciously, web scraping proves to be a valuable resource for businesses, researchers, and analysts in harnessing the wealth of information available on the internet.