Swetha. K

Rollno:235229143

Labsheet13:Retrieving Data From Web and Parsing

Question1:

Retieve data from web page using urllib and print the frequency of words from that page.

In [2]:
```
import urllib.request
from bs4 import BeautifulSoup
from collections import Counter
import re

# Define the URL of the web page you want to scrape
url = "https://www.w3schools.com/"

# Function to fetch the web page content and parse it
def fetch_and_parse(url):
    try:
        # Fetch the web page content
        response = urllib.request.urlopen(url)
        html_content = response.read()

        # Parse the HTML content using BeautifulSoup
        soup = BeautifulSoup(html_content, "html.parser")

        # Extract the text from the HTML content
        text = soup.get_text()

        return text
    except Exception as e:
        print("An error occurred:", str(e))
        return None

# Function to calculate word frequencies and print them
def calculate_word_frequencies(text):
    if text is None:
        return

    # Tokenize the text into words using regular expressions
    words = re.findall(r'\w+', text.lower())

    # Calculate word frequencies using Counter
    word_freq = Counter(words)

    # Print the word frequencies
    for word, freq in word_freq.most_common():
        print(f"{word}: {freq}")

if __name__ == "__main__":
    # Fetch and parse the web page content
    web_page_text = fetch_and_parse(url)

    # Calculate and print word frequencies
    calculate_word_frequencies(web_page_text)
```

```
learn: 91
tutorial: 85
certificate: 47
reference: 41
course: 39
exercise: 34
a: 33
web: 31
css: 30
quiz: 30
to: 29
and: 28
html: 26
c: 20
w3schools: 18
javascript: 17
your: 17
code: 16
our: 16
```

Question2.

Retrieve and display all hyperlinks(ie.,HREF attribute) from a webpage using BeautifulSoup

In [4]:
```
import urllib.request
from bs4 import BeautifulSoup

# Define the URL of the web page you want to scrape
url = "https://www.w3schools.com/"

# Function to fetch the web page content and parse it
def fetch_and_parse(url):
    try:
        # Fetch the web page content
        response = urllib.request.urlopen(url)
        html_content = response.read()

        # Parse the HTML content using BeautifulSoup
        soup = BeautifulSoup(html_content, "html.parser")

        return soup
    except Exception as e:
        print("An error occurred:", str(e))
        return None

# Function to find and display all hyperlinks (HREF attributes)
def display_hyperlinks(soup):
    if soup is None:
        return

    # Find all anchor (a) tags in the HTML
    anchor_tags = soup.find_all("a")

    # Loop through the anchor tags and display the HREF attributes
    for tag in anchor_tags:
        href = tag.get("href")
        if href:
            print(href)

if __name__ == "__main__":
    # Fetch and parse the web page content
    parsed_html = fetch_and_parse(url)

    # Display all hyperlinks
    display_hyperlinks(parsed_html)
```

https://www.w3schools.com (https://www.w3schools.com)
javascript:void(0)
javascript:void(0)
javascript:void(0)
javascript:void(0)
javascript:void(0)
javascript:void(0);
https://profile.w3schools.com/log-in?redirect_url=https%3A%2F%2Fmy-learning.w3schools.com (https://profile.w3schools.com/log-in?redirect_url=https%3A%2F%2Fmy-learning.w3schools.com)
/signup/index.php
https://profile.w3schools.com/log-in?redirect_url=https%3A%2F%2Fmy-learning.w3schools.com (https://profile.w3schools.com/log-in?redirect_url=https%3A%2F%2Fmy-learning.w3schools.com)
https://my-learning.w3schools.com (https://my-learning.w3schools.com)
https://campus.w3schools.com/collections/certifications (https://campus.w3schools.com/collections/certifications)
/spaces/index.php
https://billing.w3schools.com/products/spaces?frequency=monthly&changePlan=t

Question3:

Create a HTML file for the following students Marks and print the number of students and their names and marks.

ID	Name	mark1	mark2	mark3
DS01	Rex	87	57	74
DS02	Peter	68	98	55

In [6]:
```
# Define the student data
students_data = [
    {"ID": "DS01", "Name": "Rex", "Mark1": 87, "Mark2": 57, "Mark3": 74},
    {"ID": "DS02", "Name": "Peter", "Mark1": 68, "Mark2": 98, "Mark3": 55},
]

# Create an HTML table
html_table = """
<!DOCTYPE html>
<html>
<head>
    <title>Student Marks</title>
</head>
<body>
    <h1>Student Marks</h1>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Mark1</th>
            <th>Mark2</th>
            <th>Mark3</th>
        </tr>
"""

# Loop through the student data and add rows to the table
for student in students_data:
    html_table += f"""
        <tr>
            <td>{student["ID"]}</td>
            <td>{student["Name"]}</td>
            <td>{student["Mark1"]}</td>
            <td>{student["Mark2"]}</td>
            <td>{student["Mark3"]}</td>
        </tr>
"""

# Close the HTML table
html_table += """
    </table>
</body>
</html>
"""

# Print the number of students and the HTML table
num_students = len(students_data)
print(f"Number of students: {num_students}")
print(html_table)
```

```
Number of students: 2

<!DOCTYPE html>
<html>
<head>
    <title>Student Marks</title>
</head>
<body>
    <h1>Student Marks</h1>
    <table border="1">
        <tr>
            <th>ID</th>
            <th>Name</th>
            <th>Mark1</th>
            <th>Mark2</th>
            <th>Mark3</th>
        </tr>

        <tr>
            <td>DS01</td>
            <td>Rex</td>
            <td>87</td>
            <td>57</td>
            <td>74</td>
        </tr>

        <tr>
            <td>DS02</td>
            <td>Peter</td>
            <td>68</td>
            <td>98</td>
            <td>55</td>
        </tr>

    </table>
</body>
</html>
```

Question4:

Create a JSON file for the following Students Marks and Print the Number of students and their names and marks.

In [1]:
```
import json

# Define a dictionary with student names and marks
student_marks = {
    "Alice": 92,
    "Bob": 78,
    "Charlie": 85,
    "David": 95,
    "Eve": 88
}

# Calculate the number of students
num_students = len(student_marks)

# Create a JSON file with the student data
with open("student_marks.json", "w") as json_file:
    json.dump(student_marks, json_file)

# Read the JSON file and print the information
with open("student_marks.json", "r") as json_file:
    data = json.load(json_file)
    print(f"Number of students: {num_students}")
    for student, marks in data.items():
        print(f"Student: {student}, Marks: {marks}")
```

```
Number of students: 5
Student: Alice, Marks: 92
Student: Bob, Marks: 78
Student: Charlie, Marks: 85
Student: David, Marks: 95
Student: Eve, Marks: 88
```

Question5:

Crawl Weather of a city and Display 7 Day Forecast
Find weather data of Tiruchirappalli city from some website such as www.weather.com
Exploring page structure with chrome DevTools
Extract Information From Web Page and display the Weather forecast for 10 days or 7 days

In [3]:
```
import requests
from bs4 import BeautifulSoup

# Define the URL of the weather page for Tiruchirappalli
url = "https://weather.com/weather/7day/l/Tiruchirapalli+TN+INXX0292:1:IN"

# Send an HTTP GET request to the URL
response = requests.get(url)

# Check if the request was successful
if response.status_code == 200:
    # Parse the HTML content of the page using BeautifulSoup
    soup = BeautifulSoup(response.content, 'html.parser')

    # Locate the element containing the 7-day weather forecast data
    forecast_container = soup.find("div", class_="DailyForecast--DisclosureList

    # Extract and display the weather forecast for each day
    for day in forecast_container.find_all("div", class_="Disclosure--Disclosur
        date = day.find("span", class_="DetailsSummary--daypartDate--1Mebr").ge
        description = day.find("div", class_="DetailsSummary--dayDescription--1
        high_temp = day.find("span", class_="DetailsSummary--highTempValue--3x6
        low_temp = day.find("span", class_="DetailsSummary--lowTempValue--1DlJK

        print(f"Date: {date}")
        print(f"Description: {description}")
        print(f"High Temperature: {high_temp}")
        print(f"Low Temperature: {low_temp}")
        print()
else:
    print("Failed to retrieve the weather data.")
```

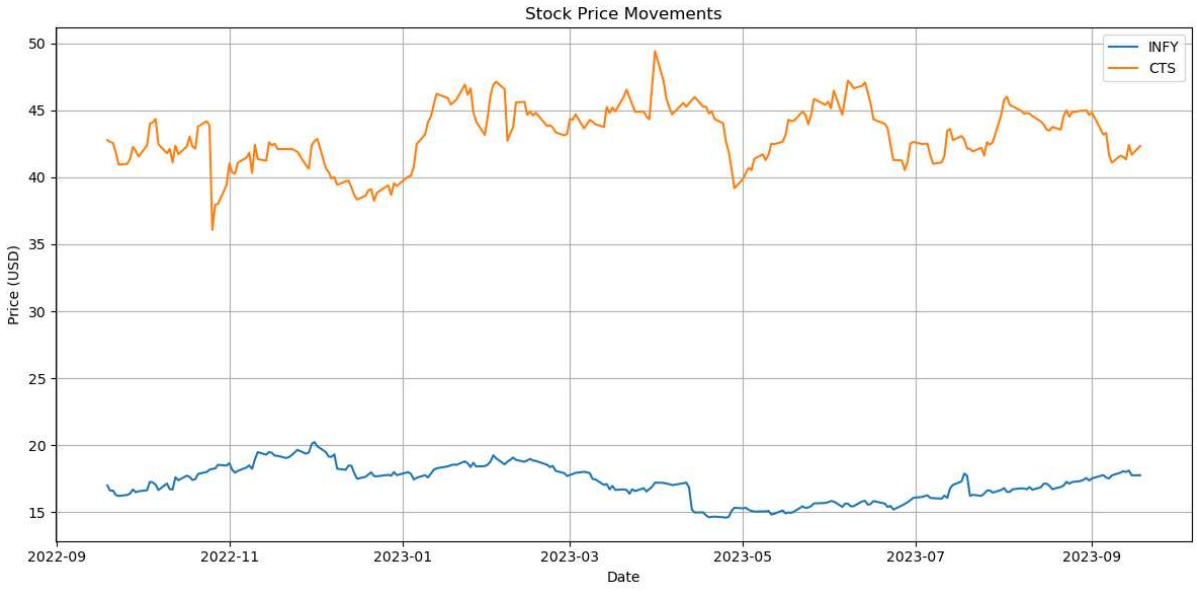
```
Failed to retrieve the weather data.
```

Question6:

Real Time Prices Crawling and Display of a specified Company
Crawl prices of a stock such as infoSys(Stock Code INFY) for a Period o
f month or year.
Plot a line graph of monthly or yearly price movements
Crawl prices of pne more stock such as CTS
Update your line graph with the prices movement of two stocks

In [8]:
```
import yfinance as yf
import matplotlib.pyplot as plt

# Define the stock symbols
stock_symbols = ["INFY", "CTS"]

# Fetch historical stock data for the past year
start_date = "2022-09-19"  # Specify the start date
end_date = "2023-09-19"    # Specify the end date

# Create an empty DataFrame to store the stock data
stock_data = {}

# Fetch stock data for each symbol
for symbol in stock_symbols:
    stock_data[symbol] = yf.download(symbol, start=start_date, end=end_date)

# Plot the stock prices
plt.figure(figsize=(12, 6))

for symbol, data in stock_data.items():
    plt.plot(data.index, data["Adj Close"], label=symbol)

# Customize the plot
plt.title("Stock Price Movements")
plt.xlabel("Date")
plt.ylabel("Price (USD)")
plt.legend()

# Show the plot
plt.grid(True)
plt.tight_layout()
plt.show()
```

```
[**********100%%**********]  1 of 1 completed
[**********100%%**********]  1 of 1 completed
```



In [ ]: