الاسم : احمد السيد عبده على فرج | سكشن 1

Chapter 9

1. Which Python library is best suited for sending HTTP requests?

a) os

b) requests

c) selenium

d) re

2. Which function in requests is used to fetch a webpage?

a) requests.open()

b) requests.page()

c) requests.get()

d) requests.read()

3. In BeautifulSoup, which method is used to extract all tags?

a) soup.find("a")

b) soup.select("a")

c) soup.find_all("a")

d) soup.get("a")

4. What does the .text property of a BeautifulSoup element return?

a) The HTML tags

b) The attribute values

c) The inner text of the tag

d) None of the above

5. Which library is used to automate interaction with JavaScript-heavy websites?

a) requests

b) BeautifulSoup

c) re

d) Selenium

6. Which of the following is an ethical consideration in web scraping?

a) Ignoring robots.txt

b) Scraping sensitive/private data

c) Respecting rate limits

d) Stealing copyrighted material

7. The compile() function is used in scraping to:

a) Convert Python code into machine language

b) Compile JavaScript on a page

c) Compile HTML into text

d) None of the above

Chapter 9 — True/False Questions

1. *requests.get() returns both the HTML source and status code.(T)*
2. *BeautifulSoup can directly fetch web pages from the internet.(F)*
3. *Selenium can be used to fill forms and click buttons on web pages.(T)*
4. *Scraping a website too frequently can overload the server.(T)*
5. *5. Saving data into JSON format requires the csv module.(F)*

Chapter 9 — Short Answer Questions

1. *Explain the difference between requests and Selenium in web scraping.*

Requests: is a lightweight HTTP library for fetching static HTML content.

Selenium :is a browser automation tool that can handle JavaScript-rendered pages, user interactions, and dynamic content by controlling a real browser.

2. *What is the purpose of the robots.txt file on a website?*
   robots.txt : is a file that tells web scrapers which parts of the site they are allowed or not to access.
3. *Write the difference between .find() and .find_all() methods in BeautifulSoup.*
   .find() : returns the first matching element.
   .find_all(): returns a list of all matching elements.

4. *Why is it important to use headers like "User-Agent": "Mozilla/5.0" in requests.get()? Some websites block requests from scripts or unknown user agents. Setting a common browser User-Agent helps mimic human browsing and avoid being blocked.*
5. *List three possible formats to store scraped data.*
   Answer:

   CSV, JSON, SQLite database (or any relational database)

Programming Problems

1. *Fetch a Web Page Title.*
   solution

```
import requests
from bs4 import BeautifulSoup

url = "https://example.com"
response = requests.get(url)

if response.status_code == 200:
    soup = BeautifulSoup(response.text, 'html.parser')
    title = soup.title.string if soup.title else "No
title found"
    print(f"Page Title: {title}")
else:
    print(f"Failed to fetch page. Status code:
{response.status_code}")
```

2. *Extract All Links Write a Python program to extract and print all links (tags with href) from the page https://example.com.*
   Solution

```
import requests
from bs4 import BeautifulSoup

url = "https://example.com"
response = requests.get(url)

if response.status_code == 200:
    soup = BeautifulSoup(response.text, 'html.parser')
    links = soup.find_all('a', href=True)

    if links:
        for link in links:
```

```
            href = link['href']
            print(href)
    else:
        print("No links found.")
else:
    print(f"Failed to fetch page. Status code:
{response.status_code}")
```

3. *Extract a Table.*

   solution

```
from bs4 import BeautifulSoup

html = """
<table>
    <tr><th>Name</th><th>Age</th></tr>
    <tr><td>Alice</td><td>25</td></tr>
    <tr><td>Bob</td><td>30</td></tr>
</table>
"""

soup = BeautifulSoup(html, 'html.parser')
rows = soup.find_all('tr')

for row in rows:
    cells = [cell.get_text(strip=True) for cell in
row.find_all(['th', 'td'])]
    print(cells)
```

4. *Automate Google Search.*

   solution

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.common.by import By
import time

driver = webdriver.Chrome()
driver.get("https://www.google.com")

search_box = driver.find_element(By.NAME, "q")
search_box.send_keys("Python Web Scraping")
search_box.send_keys(Keys.RETURN)

time.sleep(2)
print("Page Title:", driver.title)

driver.quit()
```

5. *Save Scraped Data to CSV*
   solution

```
from bs4 import BeautifulSoup
import csv

html = """
<ul>
    <li>Apple</li>
    <li>Banana</li>
    <li>Cherry</li>
</ul>
"""

soup = BeautifulSoup(html, 'html.parser')
fruits = [li.get_text(strip=True) for li in
soup.find_all('li')]

with open('fruits.csv', 'w', newline='', encoding='utf-
8') as csvfile:
    writer = csv.writer(csvfile)
    writer.writerow(['Fruit'])
    for fruit in fruits:
        writer.writerow([fruit])

print("Saved to fruits.csv")
```