

Web Scraping Task

Task Description:

You are tasked to perform web scraping on a provided HTML page that contains different types of elements. The goal is to extract specific data from the page and process it into structured formats such as CSV or JSON.

<https://baraasalout.github.io/test.html>

Practical 7:30 to 8:00

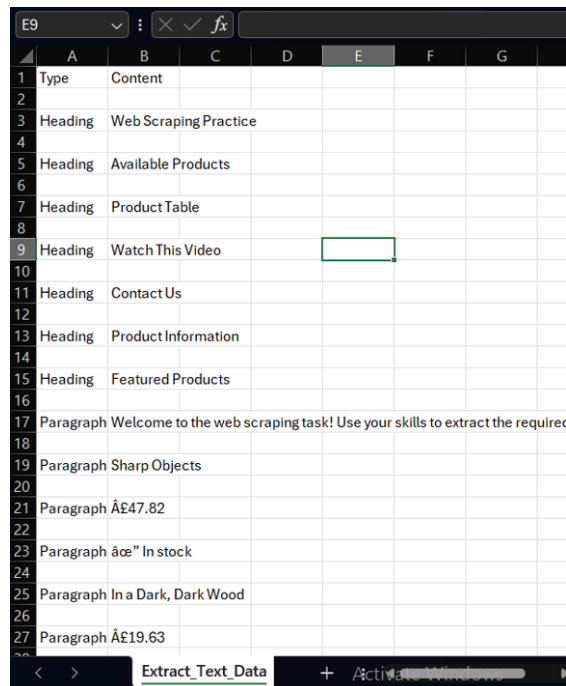
Break 8:00 to 8:20

Practical 8:20 to 8:50

Steps to Complete the Task

1. Extract Text Data:

- Extract all headings (<h1>, <h2>).
- Extract all text content inside <p> and tags
- Save this data into a **Extract_Text_Data.CSV** file.
- <https://www.pythontutorial.net/python-basics/python-write-csv-file/>



	A	B	C	D	E	F	G
1	Type	Content					
2							
3	Heading	Web Scraping Practice					
4							
5	Heading	Available Products					
6							
7	Heading	Product Table					
8							
9	Heading	Watch This Video					
10							
11	Heading	Contact Us					
12							
13	Heading	Product Information					
14							
15	Heading	Featured Products					
16							
17	Paragraph	Welcome to the web scraping task! Use your skills to extract the required					
18							
19	Paragraph	Sharp Objects					
20							
21	Paragraph	£47.82					
22							
23	Paragraph	â€” In stock					
24							
25	Paragraph	In a Dark, Dark Wood					
26							
27	Paragraph	£19.63					

2. Extract Table Data:

- Extract data from the table, including:
 - Product Name.
 - Price.
 - Stock Status.
- Save this data into a **Extract_Table_Data.CSV** file.
- <https://www.pythontutorial.net/python-basics/python-write-csv-file/>

	A	B	C	D	E
1	product	price	inStock		
2	Laptop	\$1000	Yes		
3	Smartphor	\$800	No		
4	Tablet	\$500	Yes		
5					
6					
7					

3. Extract Product Information (Cards Section):

- Extract data from the book cards at the bottom of the page, including:
 - Book Title.
 - Price.
 - Stock Availability.
 - Button text (e.g., "Add to basket").
- Save the data into a **Product_Information.JSON** file.
- <https://www.geeksforgeeks.org/how-to-convert-python-dictionary-to-json/>

4. Extract Form Details:

- Extract all input fields from the form, including:
 - Field name (e.g., username, password).
 - Input type (e.g., text, password, checkbox, etc.).
 - Default values, if any.
 - Save the data into a JSON file.
 - <https://www.geeksforgeeks.org/how-to-convert-python-dictionary-to-json/>

5. Extract Links and Multimedia:

- Extract the hyperlink (<a> tag) and its href value.
- Extract the video link from the <iframe> tag.
- Save the data into a JSON file.
- <https://www.geeksforgeeks.org/how-to-convert-python-dictionary-to-json/>

6. Scraping Challenge:

Students must write a script to extract data from the **Featured Products** section with the following requirements:

- Product Name: Located within ``.
- Hidden Price: Located within ``, which has `style="display: none;"`.
- Available Colors: Located within ``.
- Product ID: The value stored in the `data-id` attribute.
- **Example Output:**

```
[  
  
    {'id': '101', 'name': 'Wireless Headphones', 'price': '$49.99', 'colors': 'Black, White, Blue'},  
  
    {'id': '102', 'name': 'Smart Speaker', 'price': '$89.99', 'colors': 'Grey, Black'},  
  
    {'id': '103', 'name': 'Smart Watch', 'price': '$149.99', 'colors': 'Black, Silver, Gold'}  
  
]
```

Deliverables:

1. Extracted Data:
 - CSV file containing the table data.
 - JSON file containing data from the book cards and headings.
2. Code Script:
 - Provide the Python script used for scraping (comment your code for clarity).
3. Documentation:
 - Write a simple explanation of your approach, tools used, and any challenges faced.
4. Upload files on your GitHub and submit the repo link.