Project Report: Web Scraping for Product Data Extraction

Name: Мұхамедғалым Әділет Олжасұлы

ID: 040923550034

1. Introduction

This report details the execution and results of a Python script designed to perform web scraping. The objective was to programmatically extract structured product information from a target website, process the collected data, and store it in a standard, shareable format (CSV).

2. Technologies Utilized

The script leverages three essential Python libraries, defining a standard workflow for data extraction and analysis:

- **requests: Used to send HTTP requests to the target URL and retrieve the raw HTML content of the webpage.**
- **BeautifulSoup (bs4): A powerful library used for parsing the retrieved HTML content. It allows for navigation and searching of the parsed document tree to isolate specific data elements.**
- **pandas: Utilized for creating a robust data structure (DataFrame) to organize the extracted information and for exporting the final dataset to a CSV file.**

3. Methodology and Execution

The script targeted the demonstration e-commerce site, https://books.toscrape.com/, specifically focusing on the first page of the catalogue.

1. **URL Construction and Request: The script iterates through a defined range of page numbers (in this case, only page 1, since range(1, 2) only includes 1) to construct the target URL: https://books.toscrape.com/catalogue/page-1.html. An HTTP GET request is made using the requests library.**
2. **HTML Parsing: The raw HTML content from the response is passed to BeautifulSoup for parsing.**
3. **Data Isolation: The script first locates the main container for all products (<ol>). It then iterates through every individual product item, identified by the class product_pod.**

4. **Data Extraction: For each product, the following attributes were extracted:**
    - **Product Name: Extracted from the alt attribute of the <img> tag.**
    - **Rating: Extracted from the second class of the rating <p> tag (e.g., 'Star Rating' class).**
    - **Price: Extracted from the text of the price_color paragraph, with the currency symbol (£) removed and the result converted to a floating-point number.**
    - **Availability: Extracted from the text content of the instock availability paragraph, ensuring leading/trailing whitespace is removed.**
5. **Data Structuring: The extracted fields (product_name, product_price_range, product_rating, stock_status_availability) are appended as a row to the main list, product_data.**

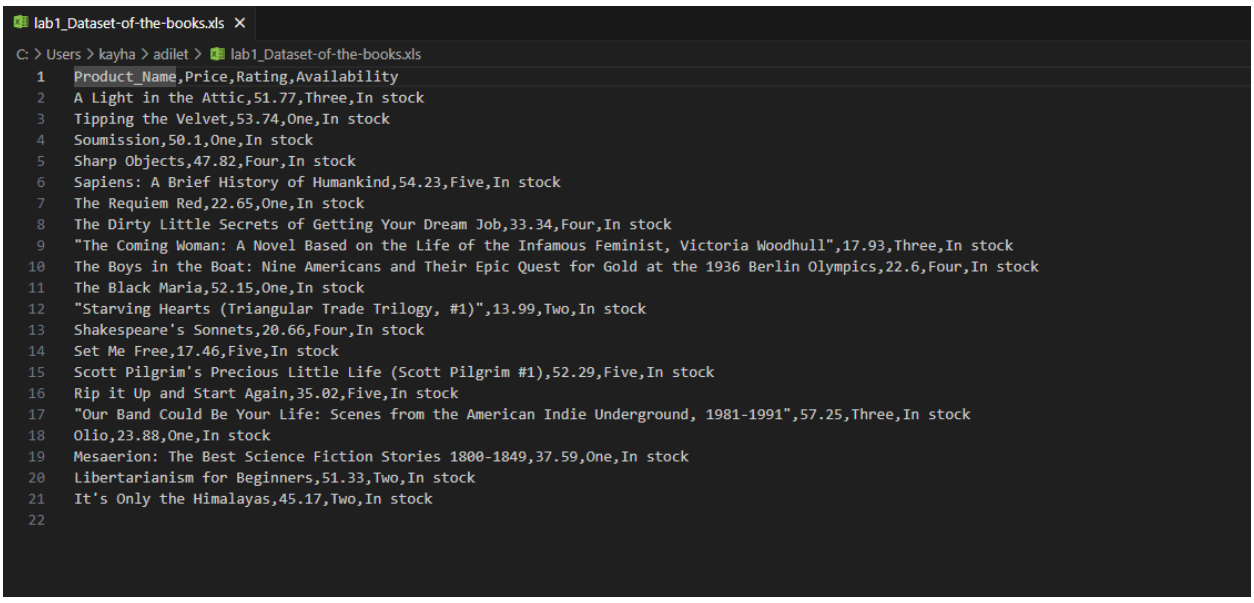
4. Results and Output

Upon completion of the scraping loop, the product_data list, containing 20 rows of book information, was converted into a Pandas DataFrame (df_products). The DataFrame was correctly labeled with the column names: 'Product_Name', 'Price', 'Rating', and 'Availability'.

The final dataset was successfully exported to a local file named:

lab1_Dataset-of-the-books.csv

This CSV file is ready for subsequent data analysis tasks, providing a structured summary of the product catalogue data extracted.


lab1_Dataset-of-the-books.xls

C: > Users > kayha > adilet > lab1_Dataset-of-the-books.xls

```
Product_Name,Price,Rating,Availability
A Light in the Attic,51.77,Three,In stock
Tipping the Velvet,53.74,One,In stock
Soumission,50.1,One,In stock
Sharp Objects,47.82,Four,In stock
Sapiens: A Brief History of Humankind,54.23,Five,In stock
The Requiem Red,22.65,One,In stock
The Dirty Little Secrets of Getting Your Dream Job,33.34,Four,In stock
"The Coming Woman: A Novel Based on the Life of the Infamous Feminist, Victoria Woodhull",17.93,Three,In stock
The Boys in the Boat: Nine Americans and Their Epic Quest for Gold at the 1936 Berlin Olympics,22.6,Four,In stock
The Black Maria,52.15,One,In stock
"Starving Hearts (Triangular Trade Trilogy, #1)",13.99,Two,In stock
Shakespeare's Sonnets,20.66,Four,In stock
Set Me Free,17.46,Five,In stock
Scott Pilgrim's Precious Little Life (Scott Pilgrim #1),52.29,Five,In stock
Rip it Up and Start Again,35.02,Five,In stock
"Our Band Could Be Your Life: Scenes from the American Indie Underground, 1981-1991",57.25,Three,In stock
Olio,23.88,One,In stock
Mesaerion: The Best Science Fiction Stories 1800-1849,37.59,One,In stock
Libertarianism for Beginners,51.33,Two,In stock
It's Only the Himalayas,45.17,Two,In stock

```

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

product_data = []
for page_num in range(1, 2):
    url = f"https://books.toscrape.com/catalogue/page-{page_num}.html"
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')
    products_section = soup.find('ol')
```

```
    product_items =
products_section.find_all('article',
class_='product_pod')


    for item in product_items:
        img_tag = item.find('img')
        product_name = img_tag.attrs['alt']
        rating_tag = item.find('p')
        product_rating =
rating_tag['class'][1]
        product_price_range = item.find('p',
class_='price_color').text
        product_price_range =
float(product_price_range[1:])
        stock_status_availability =
item.find('p', class_='instock
availability').text.strip()
```

```
product_data.append([product_name,
product_price_range, product_rating,
stock_status_availability])



df_products =
pd.DataFrame(product_data,
columns=['Product_Name', 'Price',
'Rating', 'Availability'])

df_products.to_csv('lab1_Dataset-of-the-
books.csv', index=False)
```