**ASSESSMENT DOCUMENTATION**

**1. Web Scraping:**

**(A) Tools and Libraries Used:**

🡪 Python

🡪 Beautiful Soup (bs4) - for HTML parsing and traversing

🡪 Requests - for making HTTP request.

🡪 We used Python with Beautiful Soup and Requests to access and scrape the webpage's HTML, extract the relevant information, and structure it into the required data format**.**

**2. Challenges Faced and How They Were Overcome:**

**Challenge 1: Web page structure changes:**

Web pages can be updated or modified, leading to changes in the HTML structure. To mitigate this, we regularly inspected the webpage structure and made necessary adjustments to the scraping code accordingly.

**Challenge 2: Handling dynamic content:**

Some websites load content dynamically using JavaScript, making it challenging to scrape using traditional methods. In this case, since the webpage didn't rely heavily on JavaScript, standard HTTP requests and Beautiful Soup were sufficient to retrieve the data.

**Challenge 3: Data formatting inconsistencies:**

Data extracted may have inconsistencies or unexpected formatting. We handled this by incorporating appropriate data cleaning and formatting procedures to ensure the extracted data meets the required format.

**3. Assumptions Made During the Process:**

The webpage structure and class names would remain consistent during the scraping process. Any changes in these might require modifications in the scraping code.

The TRAVELS (X/Y/Z) format for each product will always be in the specified structure. If the structure changes, the code may need adjustments.

The product specifications and details on the webpage are accurate and up to date at the time of scraping.

By following these steps and guidelines, we successfully scraped the specified data from the provided webpage and structured it as required.