Web Scraping Report: Mytheresa Website

1. Introduction

Web scraping is the process of automatically extracting data from websites. This report explains the scraping of product details from the Mytheresa website, which sells luxury fashion items. The goal was to collect product information, such as names, prices, and sizes.

2. Objective

The main aim of this project was to:

- Scrape details of shoes for men from the Mytheresa website.
- Extract important data like product name, price, brand, and image URL.
- Handle pagination to get data from multiple pages.

3. Tools Used

The following tools were used for scraping:

- **Scrapy**: A Python tool for extracting data from websites.
- **Python 3**: The programming language used.
- **CSS Selectors**: Used to find specific parts of the webpage to extract data.
- **JSON/CSV**: The formats used to save the extracted data.

4. Scrapy Spider Setup

A Scrapy spider was created to scrape product data from Mytheresa. Here's how it works:

- **Start URL**: The spider begins on the page listing men's shoes: https://www.mytheresa.com/int/en/men/shoes?rdr=mag
- **Parsing the Listing Page**: The spider extracts links to individual product pages.
- **Handling Pagination**: The spider follows the "next" page links to scrape products from all pages.
- **Extracting Product Data**: On each product page, the spider collects the following details:
    - **breadcrumbs**: Navigation trail to understand where the product is located.
    - **image_url**: The main image of the product.
    - **brand**: The product's brand name.
    - **product_name**: The name of the product.

- **price**: The current price of the product.
- **reviews**: If available, the product's review score.
- **colour**: The color of the product.
- **sizes**: Available sizes for the product.
- **description**: A brief description of the product.
- **sku**: Stock-keeping unit for tracking.
- **product_id**: Unique product identifier.

5. Process of Scraping

Here is how the data was collected:

1. **Scrape Listing Page**: The spider starts on the men's shoes page and finds product links.
2. **Follow Product Links**: For each product, the spider clicks the link and goes to the product detail page.
3. **Extract Product Information**: The spider extracts details like price, brand, and size from each product page.
4. **Handle Pagination**: After scraping a page, the spider follows the "next" page link to continue scraping.

6. Results

After scraping, we got:

- **Number of Products**: Data was successfully collected from multiple pages, including details like price, brand, and size.
- **Data Format**: The data was saved in both JSON and CSV formats for analysis.

8. Conclusion

The web scraping project was successful in collecting product information from the Mytheresa website. The Scrapy spider navigated the website, followed pagination links, and extracted the necessary data. Although some challenges like dynamic content and anti-scraping measures were encountered, they were manageable.

9. Submission

The code has been submitted via a private GitHub repository.

HTTP Urls: https://github.com/Kannanpbinu/myth.git

SSH Urls: git@github.com:Kannanpbinu/myth.git

The output files (CSV and JSON) are available for download via the Dropbox link.
Urls for CSV
File:https://www.dropbox.com/scl/fo/kgqjceq8ephyih1ny0woo/AF9JUc0nfYjf4RW5hH71SdU?rlkey=qp2egx4pjmeibebqaahhaekqd&st=dq5btpqw&dl=0

Urls for JSON Files:
https://www.dropbox.com/scl/fo/paxnt4c88dbl0l78u0hqe/AEIqhCn0-V844iPk1-Cq2h4?rlkey=cxck9wgcefn5u52bfs7w1xn4v&st=rfwfpvfq&dl=0