1. **Thế nào là Crawl Data**

Crawl data hay còn gọi là cào dữ liệu, là quá trình thu thập dữ liệu và thông tin website nhằm phục vụ nhiều mục tiêu khác nhau. Theo đó, các bot của công cụ tìm kiếm (Search Engine) như Google, Bing,… sẽ lần lượt truy cập vào tất cả trang trên website cũng như liên kết liên quan để thống kê dữ liệu.

1. **Tìm hiểu các kỹ thuật: Beauty Soup, Scrapy và Selenium**

**Beautiful Soup**

* Beautiful Soup (BS4) là một thư viện phân tích cú pháp có thể sử dụng các parsers khác nhau từ đó có thể trích xuất dữ liệu từ các tài liệu HTML và XML một cách dễ dàng. Về mặc định, Beautiful Soup sử dụng parser cơ bản của Python. Mặc dù khá linh hoạt và dễ sử dụng, parser này là có hiệu năng khá kém do tốc độ xử lý khá chậm. Tin tốt là bạn hoàn toàn có thể hoán đổi trình phân tích cú pháp của nó bằng một trình phân tích cú pháp nhanh hơn nếu bạn cần cải thiện nhiều tốc độ của ứng dụng.
* Sau khi phân tích các HTML cũng như XML đầu vào, Beautiful Soup cho phép chúng ta dễ dàng di chuyển, tìm kiếm, thay đổi cũng như trích xuất dữ liệu từ cây cú pháp. Cú pháp rõ ràng linh hoạt tương tự cách chúng ta tương tác với DOM bằng các thư viện JavaScript là một trong những lý do khiến Beautiful Soup trở thành một trong những công cụ phổ biến nhất cho việc thu thập dữ liệu web, bên cạnh sự mạnh mẽ của nó.

**Scrapy**

* Về mặt kỹ thuật, Scrapy không phải một thư viện mà là một framework phục vụ mục đích thu thập dữ liệu. Điều đó có nghĩa là bạn có thể sử dụng nó để quản lý các yêu cầu, duy trì các phiên của người dùng, theo dõi chuyển hướng và xử lý các pipelines đầu ra. Nó cũng có nghĩa là bạn có thể hoán đổi các mô-đun riêng lẻ với các thư viện duyệt web Python khác. Ví dụ: nếu bạn cần chèn Selenium để quét các trang web động, bạn có thể làm điều đó

**Selenium**

* Selenium là một trong những công cụ kiểm thử phần mềm tự động mã nguồn mở mạnh nhất hiện nay cho việc kiểm thử ứng dụng Web. Selenium script có thể chạy được trên hầu hết các trình duyệt như IE, Mozilla FireFox, Chrome, Safari, Opera; và hầu hết các hệ điều hành như Windows, Mac, Linux. Về cơ bản mà nói, quá trình scraping cũng tương tự như quá trình kiểm thử ứng dụng tự động bởi chúng đều thực hiện một chuỗi thao tác tương tác với các trang web một cách tự động và liên tục. Bởi vậy, Selenium thường xuyên được sử dụng nhất là khi cần thu thập từ các trang web SPA - thứ mà khó có thể thu thập được dữ liệu từ nó nếu như phần mã JavaScript của chúng không được thực thi.

1. **Mục đích sử dụng của các công cụ**

**Beautiful Soup:**

**Mục đích:** Beautiful Soup (BS4) là một thư viện Python dùng để parse (phân tích) dữ liệu HTML và XML. Nó thường được sử dụng để trích xuất thông tin từ trang web cụ thể.

**Selenium:**

**Mục đích:** Selenium là một công cụ tự động hóa trình duyệt web. Nó cho phép bạn mô phỏng hành vi của người dùng trên trình duyệt để thu thập dữ liệu hoặc thực hiện các tác vụ trên trang web.

**Scrapy:**

**Mục đích:** Scrapy là một framework Python mạnh mẽ dành cho việc crawl và extract dữ liệu từ trang web. Nó cung cấp một cách cấu trúc để xây dựng các spider (các chương trình crawl) và quản lý các yêu cầu và dữ liệu thu thập.

1. **Nêu sự khác biệt giữa các công cụ này trong Web Crawl Data**

**Beautiful Soup:**

* Loại công cụ: Thư viện phân tích HTML/XML.
* Sử dụng cho: Trích xuất thông tin tĩnh từ các trang web HTML/XML.
* Cách hoạt động: Phân tích cú pháp HTML để tìm và trích xuất dữ liệu.
* Ưu điểm:
* Dễ học và sử dụng.
* Thích hợp cho các trang web có dữ liệu tĩnh.
* Không cần trình duyệt web.
* Nhược điểm:
* Không thể thực hiện các tác vụ tương tác với trang web (ví dụ: click nút).
* Không hỗ trợ quy tắc crawl hoàn chỉnh.

**Scrapy:**

* Loại công cụ: Framework crawl web.
* Sử dụng cho: Crawl và extract dữ liệu từ các trang web, đặc biệt là các trang web lớn và phức tạp.
* Cách hoạt động: Tạo spider (chương trình crawl) để tự động lặp lại và trích xuất dữ liệu từ nhiều trang web.
* Ưu điểm:
* Linh hoạt và mạnh mẽ, hỗ trợ nhiều quy tắc crawl.
* Đa luồng cho hiệu suất cao.
* Có thư viện đầu ra cho nhiều định dạng dữ liệu.
* Nhược điểm:
* Cần kiến thức lập trình tốt hơn.
* Dựng cấu trúc và cấu hình phức tạp hơn so với Beautiful Soup.

**Selenium:**

* Loại công cụ: Công cụ tự động hóa trình duyệt web.
* Sử dụng cho: Crawl dữ liệu từ các trang web đòi hỏi tương tác với trình duyệt (ví dụ: điền form, click nút) hoặc trang web sử dụng JavaScript để tạo nội dung.
* Cách hoạt động: Tự động mô phỏng hành vi người dùng trên trình duyệt để crawl dữ liệu.
* Ưu điểm:
* Có thể xử lý các trang web động và có tương tác.
* Thường dễ dàng theo dõi và gỡ lỗi.
* Nhược điểm:
* Chậm hơn so với các phương pháp crawl dựa trên HTTP.
* Cần cài đặt và cấu hình trình duyệt web.
* Cần kiến thức lập trình cơ bản.

1. **Nêu ra giải pháp chọn lựa các công cụ này trong từng tính huống cụ thể**
2. **Trích xuất dữ liệu tĩnh từ trang web HTML:**

Giải pháp: **Beautiful Soup.**

Lý do: Beautiful Soup làm việc tốt cho việc parse và trích xuất dữ liệu từ HTML. Nó đủ đơn giản để sử dụng cho các trang web có dữ liệu tĩnh.

1. **Crawl dữ liệu từ nhiều trang web khác nhau:**

Giải pháp: **Scrapy.**

Lý do: Scrapy là một framework crawl web mạnh mẽ, giúp bạn xây dựng các spider để tự động crawl và trích xuất dữ liệu từ nhiều trang web khác nhau. Nó hỗ trợ đa luồng và quản lý yêu cầu tốt.

1. **Crawl dữ liệu từ trang web đòi hỏi tương tác với trình duyệt:**

Giải pháp: **Selenium.**

Lý do: Selenium cho phép bạn mô phỏng hành vi người dùng trên trình duyệt web, điều này rất hữu ích khi bạn cần thực hiện các tác vụ như điền form, click nút hoặc crawl các trang web sử dụng JavaScript để tạo nội dung động.