webscrapping

January 13, 2024

Day 84-85 Webscrapping By: Loga Aswin

Web Scraping Overview:

Web scraping is often used to automate the extraction of large amounts of data from websites for various purposes.

It involves sending HTTP requests to the server, receiving HTML content in response, and then parsing and extracting desired information.

[1]:
```
# Sample code
import requests

from bs4 import BeautifulSoup

req = requests.get("https://www.google.com/")

soup = BeautifulSoup(req.content, "html.parser")

print(soup.get_text())
```

```
GoogleSearch Images Maps Play YouTube News Gmail Drive More »Web History |
Settings | Sign in Advanced searchAdvertisingBusiness SolutionsAbout Google©
2024 - Privacy - Terms
```

To install libraries: pip install requests beautifulsoup4

How Web Scrapers Work:

1. **HTTP Requests:** Web scrapers use the HTTP protocol to request and receive HTML content from web servers.
2. **HTML Parsing:** Parsers like BeautifulSoup or lxml parse the HTML, creating a parse tree that can be navigated to extract data.
3. **Data Extraction:** Extract specific data using CSS selectors, XPath, or other methods based on the HTML structure.
4. **Handling Dynamic Content:** Selenium automates browsers, allowing interaction with dynamically-loaded content.

Types of Web Scrapers:

1. **Static Web Scrapers:** Ideal for websites with fixed content.Requires periodic re-scraping to update information.
2. **Dynamic Web Scrapers:** Essential for websites with content loaded via JavaScript. Can interact with the page in real-time for data retrieval.
3. **API Scrapers:** Interact with APIs provided by websites for structured data access. Preferable when available to avoid HTML parsing complexities.

Libraries Used in Web Scraping:

1. **BeautifulSoup (Python):** Parses HTML and XML documents, providing a Pythonic way to navigate and search the parse tree.
2. **Requests (Python):** Simplifies sending HTTP requests and handling responses in Python.
3. **Selenium (Python):** Automates browsers for dynamic content scraping. Can be combined with BeautifulSoup for comprehensive web scraping.

Where Web Scraping is Used:

1. **Data Mining:** Extracting insights and patterns from large datasets.
2. **Price Monitoring:** Monitoring competitors' prices for dynamic pricing strategies.
3. **Content Aggregation:** Collecting data for news articles, blog posts, or research papers.
4. **Business Intelligence:** Gathering data for decision-making processes within businesses.
5. **Real Estate:** Extracting property listings, prices, and market trends.