**Group:** Isaac Adams, Owen Miller, Gavin Stone, Samuel Shevlin, Kaden Hicklin, Sullivan Gleason  
  
**Topic:** Analytical Comparison of “BeautifulSoup” and “MechanicalSoup” scraping tools

**Architecture:** Both “BeautifulSoup” and “MechanicalSoup” are useful scraping tools that exist as Python libraries. Where as “BeautifulSoup” is used in the downloading and reading of HTML and XML pages, “MechanicalSoup” is more of a hodgepodge of “BeautifulSoup” and the ability to make requests and calls to the pages themselves [1].

“BeautifulSoup” is only a scraping tool. It is designed specifically to gather and present information about a website’s HTML or XML elements and relay them to the user in a manner that is understandable to them. It gathers the information, but has no ability to act with consideration to what it obtains and acts only as a messenger to the human or entity in command of it [1]. It creates a Python structure similar to a DOM tree in order to represent the scraped data in a cohesive manner, and has many internal functions that it uses to do this. For instance, the find() and find\_all() functions are used to locate elements that match the given conditions and show the first element or all elements respectively. “BeautifulSoup” also has a few parsing libraries that can be called using the BeautifulSoup() function. These include Python’s built-in parser “html.parser”, lxml’s “lxml” and “xml” and the library “html5lib”. Each of these have benefits and downsides that determine which is best in different circumstances, but also have differences in the way they output document data too [2][3].

Some of the various “find” functions are as follows:

* find\_all()
* find()
* find\_parents()
* find\_parent()
* find\_all\_next()
* find\_next()
* find\_all\_previous()
* find\_previous()
* etc

“BeautifulSoup” also has many functions that can modify the created DOM tree structures once made. A few of these functions are:

* append()
* extend()
* NavigableString()
* insert()
* clear()
* extract()
* decompose()
* etc

“MechanicalSoup” is designed to interface with and directly act upon web pages and sites, interacting with the website based on the data that was scraped using “BeautifulSoup”. It is a superset of “BeautifulSoup” and thus contains the same abilities, albeit paired with more tools and capabilities. It is considered high-level, in that it operates in a manner consistent with a person, interacting with buttons, fields, links, and any other high-level interfaces or elements of an interface [1][4].

In addition to the functions from “BeautifulSoup”, “MechanicalSoup” also has a class to build forms called “Form” and some to raise exceptions such as the “InvalidFormMethod” and “LinkNotFountError” classes. There are also classes that handle browser state information, such as “\_BrowserState” and “StatefulBrowser” [4].

**References:**

[1] *How does mechanicalsoup differ from Beautifulsoup?*. How does MechanicalSoup differ from BeautifulSoup? | WebScraping.AI. (n.d.). <https://webscraping.ai/faq/mechanical-soup/how-does-mechanicalsoup-differ-from-beautifulsoup#:~:text=To%20summarize%2C%20MechanicalSoup%20is%20a,interacting%20with%20web%20forms%20programmatically>.

No date of publication given on the website.

[2] Singh et al. (2024, April 16). *Beautifulsoup web scraping*. Bright Data. <https://brightdata.com/blog/how-tos/beautiful-soup-web-scraping>

[3] Richardson, L. (n.d.). *Beautiful Soup documentation¶*. Beautiful Soup Documentation - Beautiful Soup 4.12.0 documentation. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

No date of publication given on website; Author is listed as the copyright holder for the website found at the bottom of the page.

[4] Moy, M. (2023, December 24). *MechanicalSoup/MechanicalSoup: A python library for automating interaction with websites.* GitHub. <https://github.com/MechanicalSoup/MechanicalSoup>

GitHub repository; publication date is the date of the most recent revision given on the project

Citations made with help from CitationMachine found on <https://citationmachine.net>