**eBay Web Scraper Bot Reflection**

During my time working on the web scraper bot, I followed a YouTube tutorial by TechWithTim on the Python library “Beautiful Soup.” This library allowed me to find a URL, store the HTML of the page, then search for keywords in the HTML. With this tutorial, I was able to apply this to a personal hobby of mine – video game collecting. I would traverse the HMTL attribute hierarchy to capture the elements of the page I wanted to find i.e. the title of a listing and the price. With this information, I wanted a way to alert myself for listings that fit the criteria that I was looking for in a video game, so I connected this information with a Discord bot and hosted the bot through Repl.it.

I then tried to apply this logic to more advanced sites such as Amazon, but shortly became aware of their web scraping prevention tactics. I became familiar with User Agents as well as ways to rotate User Agents to workaround the prevention but ended up concluding the project.

**Key takeaways from the Beautiful Soup Library**

* Generic Structure to store a URL.

*import* BeautifulSoup

*import* requests

url = ""

results = requests.get(url)

doc = BeautifulSoup(results.text, "html.parser")

* To find a class, use “class\_”

.find(["h1"], class\_="item\_title”)

**Key Takeaways from the Discord Library**

* Generic structure to create a bot (discord developer portal to obtain bot token).

import discord

token = ‘’

client = discord.Client()

//do stuff

client.run(token)